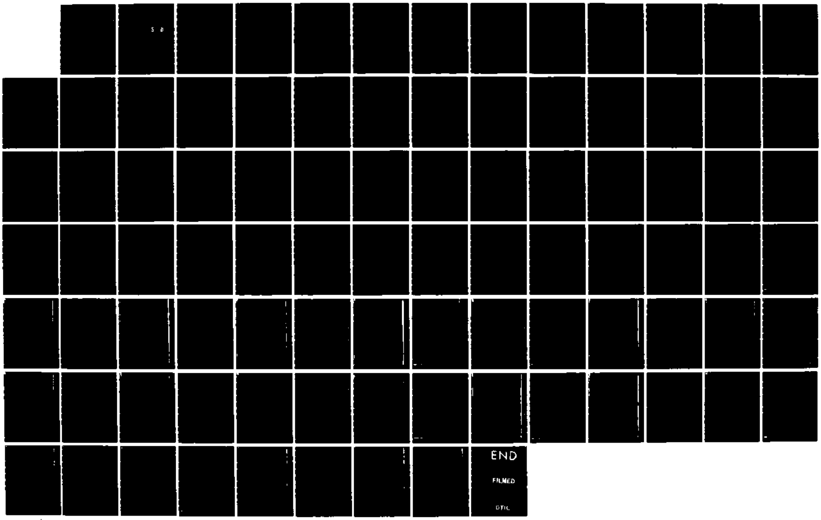
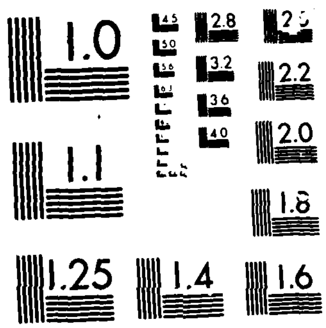
AD-A165 801 INSTRUCTION FOR USE OF THE DESADE CABLE ARRAYS STATIC DEFLECTION PROGRAM(U) NAVAL FACILITIES ENGINEERING COMMAND WASHINGTON DC CHESAPEAKE DIV H SHIH SEP 77 1/1

UNCLASSIFIED CHES/NAVFAC-FPO-7737 F/G 9/2 NL





MICROCOPY RESOLUTION TEST CHART

FPO
7737

TR-FPO-1E-23

AD-A165 801

INSTRUCTION FOR USE OF THE DESADE CABLE ARRAYS STATIC DEFLECTION PROGRAM

H. SHIH

SEPTEMBER 1977

Approved by: S.C. LING, Manager
Engineering Analysis
Branch

Approved by: C.E. BODEY, DIRECTOR
Engineering and Design
Division

OCEAN ENGINEERING AND CONSTRUCTION PROJECT OFFICE
CHESAPEAKE DIVISION
NAVAL FACILITIES ENGINEERING COMMAND
WASHINGTON, D.C. 20374

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

AD-A165801

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION
Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY

3. DISTRIBUTION AVAILABILITY OF REP.
Approved for public release;
distribution is unlimited

2b. DECLASSIFICATION/DOWNGRADING SCHEDULE

4. PERFORMING ORGANIZATION REPORT NUMBER
FPO 7737, TR-FPO-1E-23

5. MONITORING ORGANIZATION REPORT #

6a. NAME OF PERFORM. ORG.
Ocean Engineering
& Construction
Project Office
CHESNAVFACENGCOM

6b. OFFICE SYM

7a. NAME OF MONITORING ORGANIZATION

6c. ADDRESS (City, State, and Zip Code)
BLDG. 212, Washington Navy Yard
Washington, D.C. 20374-2121

7b. ADDRESS (City, State, and Zip)

8a. NAME OF FUNDING ORG.

8b. OFFICE SYM

9. PROCUREMENT INSTRUMENT INDENT #

8c. ADDRESS (City, State & Zip)

10. SOURCE OF FUNDING NUMBERS

PROGRAM ELEMENT #	PROJECT #	TASK #	WORK UNIT ACCESS #

11. TITLE (Including Security Classification)
Instruction for Use of the Descade Cable Arrays Static Deflection Program

12. PERSONAL AUTHOR(S)
H. Shih

13a. TYPE OF REPORT

13b. TIME COVERED
FROM TO

14. DATE OF REP. (YYMMDD)
9/77

15. PAGES
80

16. SUPPLEMENTARY NOTATION

17. COSATI CODES

FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if nec.)
Arrays, Cable

19. ABSTRACT (Continue on reverse if necessary & identify by block number)

This note describes the use of the DESADE program version presently running on the CDC 6600 through the FPO-1 computer termianl. A tri-moor cable array structure is given here as an example to illustrate the procedures in using this program. The original manual on which this note is based was (Con't)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT
SAME AS RPT.

21. ABSTRACT SECURITY CLASSIFICATION

22a. NAME OF RESPONSIBLE INDIVIDUAL
Jacqueline B. Riley

22b. TELEPHONE
202-433-3881

22c. OFFICE SYMBOL

DD FORM 1473, 84MAR

SECURITY CLASSIFICATION OF THIS PAGE

BLOCK 19 (Con't)

prepared by R.A. Skop, and J. mark of Naval Research Laboratory, Washington, D.C.

The program calculates the current - induced static deflections of structural cable arrays. Attached to this note are results of test runs made for the Linear Chari candidate configurations. This was designed to serve as a check for the on going Linear Chair analysis contract with DTNSRDC using the DSSM program.

TABLE OF CONTENTS

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

ABSTRACT

This note describes the use of the DESADE program version presently running on the CDC 6600 through the FPO-1 computer terminal. A tri-moor cable array structure is given here as an example to illustrate the procedures in using this program. The original manual on which this note is based was prepared by R. A. Skop and J. Mark of Naval Research Laboratory, Washington, D. C.

The program calculates the current - induced static deflections of structural cable arrays. Attached to this note are results of test runs made for the Linear Chair candidate configurations. This was designed to serve as an check for the on going Linear Chair analysis contract with DTNSRDC using the DSSM program.

1. Note of the program

Usefulness	0	computing current - induced static deflections of structural cable arrays.
	0	Most general (Ref. 1)
	0	Experimentally validated (Ref. 3)
Method	0	Use Method of imaginary reactions (Ref. 4) and method of successive approximations for treating position and configuration dependent forces (Ref. 5)
Assumptions	0	The hydrodynamic force on the devices and cables is the normal drag.
	0	The current is undirectional and horizontal (with certain modification it could accept arbitrary current field with variable directions).
Capabilities	0	arbitrarily configured arrays of up to 22 cables
	0	variable cable materials
	0	any number of discrete devices
Limitations	0	Cable segments cannot be on the ocean floor.
	0	Dimensions of discrete device must be small compared to overall array dimensions (not valid for moored submerged submarine.)
	0	All parts of the array must be submerged (unless the surfaced device coordinates are specified).

2. Array description

EXAMPLE : A sea spider (tri-moor) array structure is used here to illustrate the steps required for in preparing array structure input data. The array design is sketched in Figure 1. It is intended to calculate the static deflections and the forces of each structural member under hydrodynamic loading of an ocean current (Figure 2.)

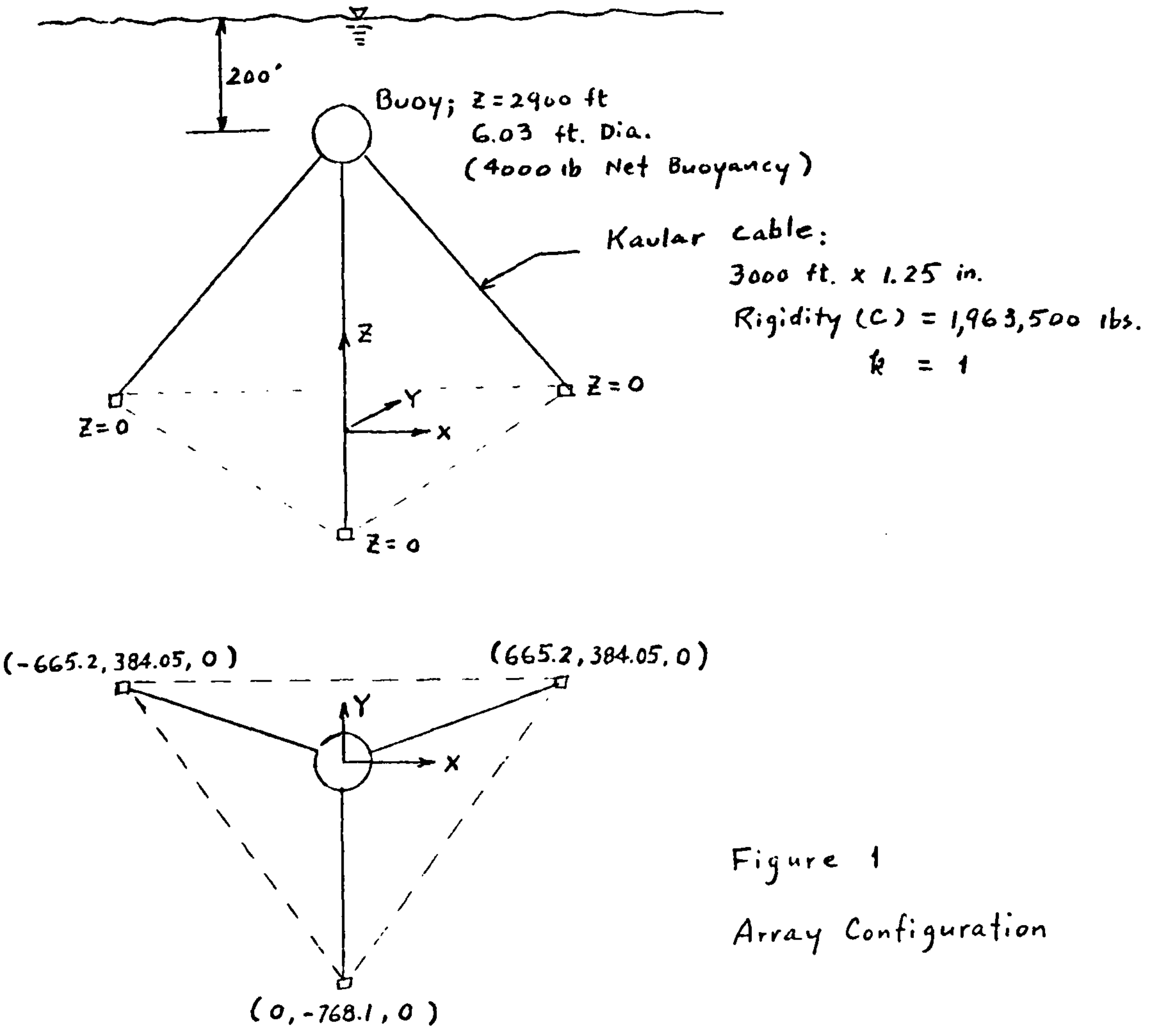
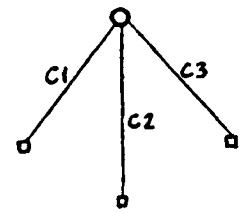


Figure 1

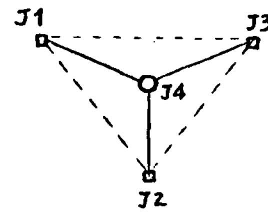
Array Configuration

STEPS

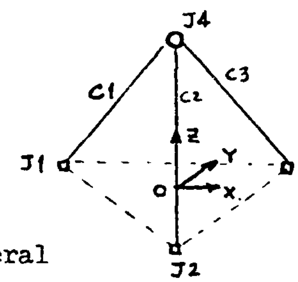
a. Number the cables consecutively from one to the total number of cables in the array. Each cable so designed must have uniform properties (weight, diameter, drag coefficient, and constitutive relation). A change in property also requires a change in cable number.



b. Number the junctions consecutively from one to the total number of junctions in the array. A junction may designate an anchor, the intersection point of two or more cables, or the free end of a cable.



c. Use a fixed, right-hand cartesian coordinate system to describe the configuration of the array in the space. The origin of the system can be arbitrarily located. Z axis is defined parallel to the direction of gravity and increasing upward. In this example the origin is placed at the geometric center of the equilateral triangle formed by the three anchors in the sea floor plan.



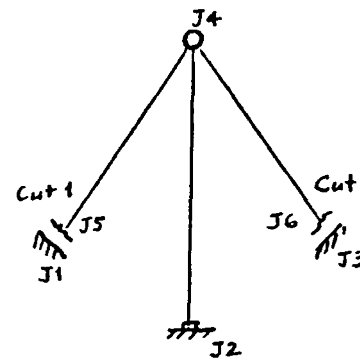
d. Tabulate the anchor coordinates (Table 1.)

Table 1

Junction No. of anchor	Anchor coordinates		
	X (ft)	Y (ft)	Z (ft)
1	-665.2	384.05	0
2	0	-768.1	0
3	665.2	384.05	0

e. Reduce the array to statically determinate array structure. The cuts are made in the following manner:

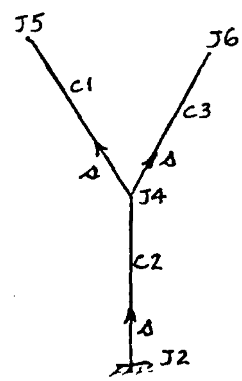
1. Number of cuts = number of cables + number of anchors - number of junctions



2. Cuts must be made at end points of cables (adjacent to junctions).
3. The first group of cuts must be made so as to release all but one cable from an anchor.
4. The remaining required cuts are made within the array and must be located so as not to break the array into separated parts.
5. Assign each new cut a consecutive junction number continuing from the last-used junction number.
6. Tabulate the junction number

Table 2

Cut number	Original Junction Number at cut	New Junction Number at cut
1	1	5
2	3	6



7. Define the directions of increasing arc length,(s), along each cable by using a topological tree diagram for the reduced array structure (see sketch). The directions are indicated by the arrowheads in the climbing 'up' direction from root to top.
8. Following the directions given in step 7 tabulate the junction number of the starting (s=0) and ending (s=L) points for each cable (Table 3).

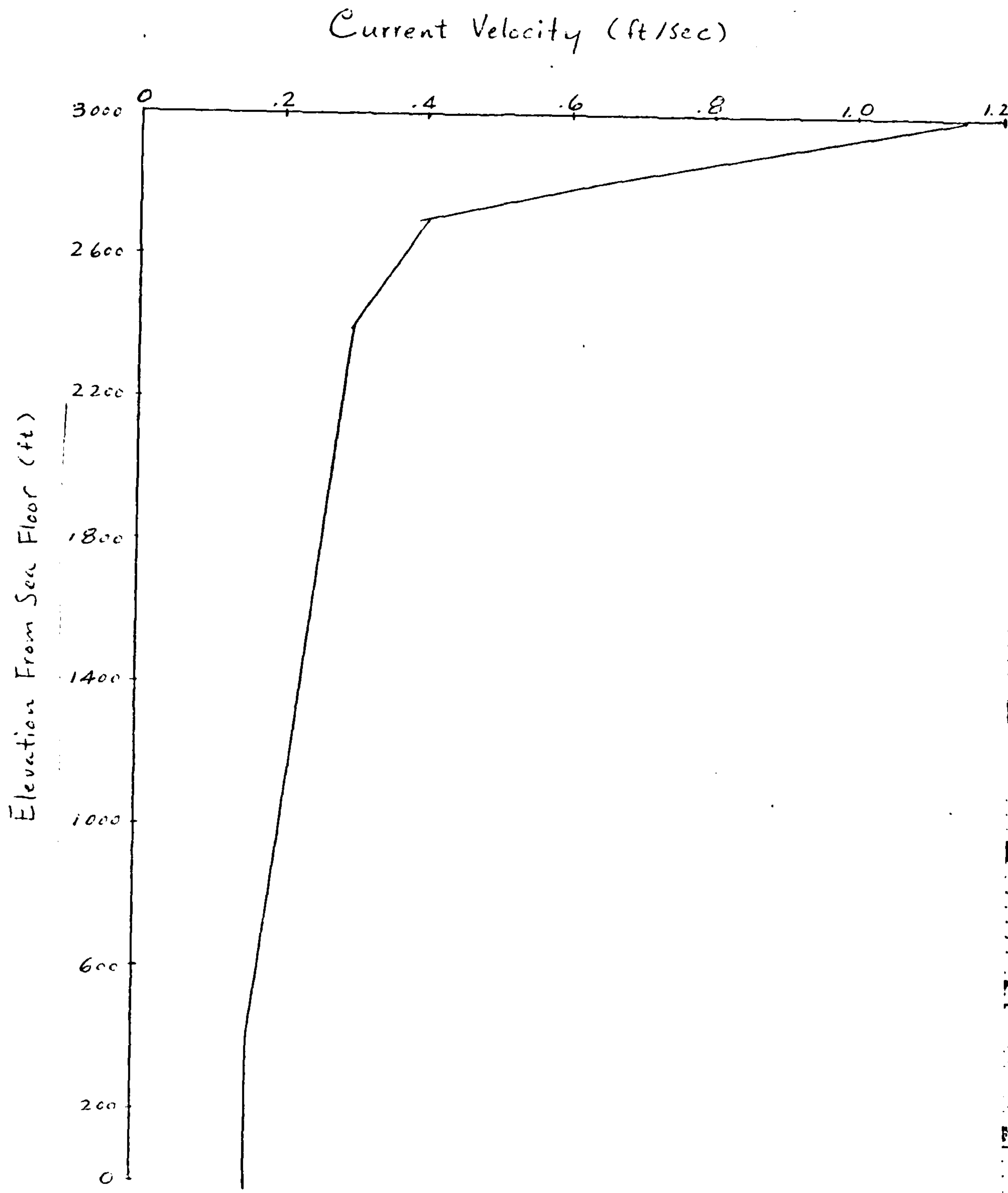


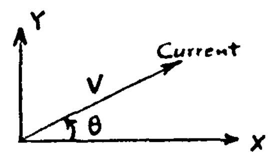
Figure 2 Current Profile in St Croix

Table 3

Cable number	Junction number at s=0	Junction number at s=L
1	4	5
2	2	4
3	4	6

The starting and ending points of each cable can be identified by following the direction of 'climbing up' the topological tree diagram. s is the arc length measured from starting junction. L is the length of each cable.

3. Current description



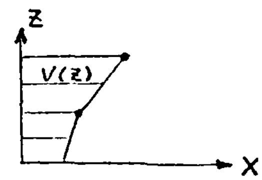
The standard current field is defined as

$$V = V(z)\ (\cos\theta\ \hat{i} + \sin\theta\ \hat{j})$$

where $v(z)$ = current value at z.

θ = current angle with respect to x axis

$\hat{i}, \hat{j}$ = unit vectors in the x and y azes



The program generates the velocity profile by connecting each current data point by a series of straight lines. The current below the minimum z value is considered the same as V(z-minimum). Using the current profile given in Figure 2 the input current data can be tabulated (Table 4.)

Table 4

Z Coordinate (ft)	V (Z) (Knots)
0	.1
400	.1
2400	.18
2700	.24
2900	.53

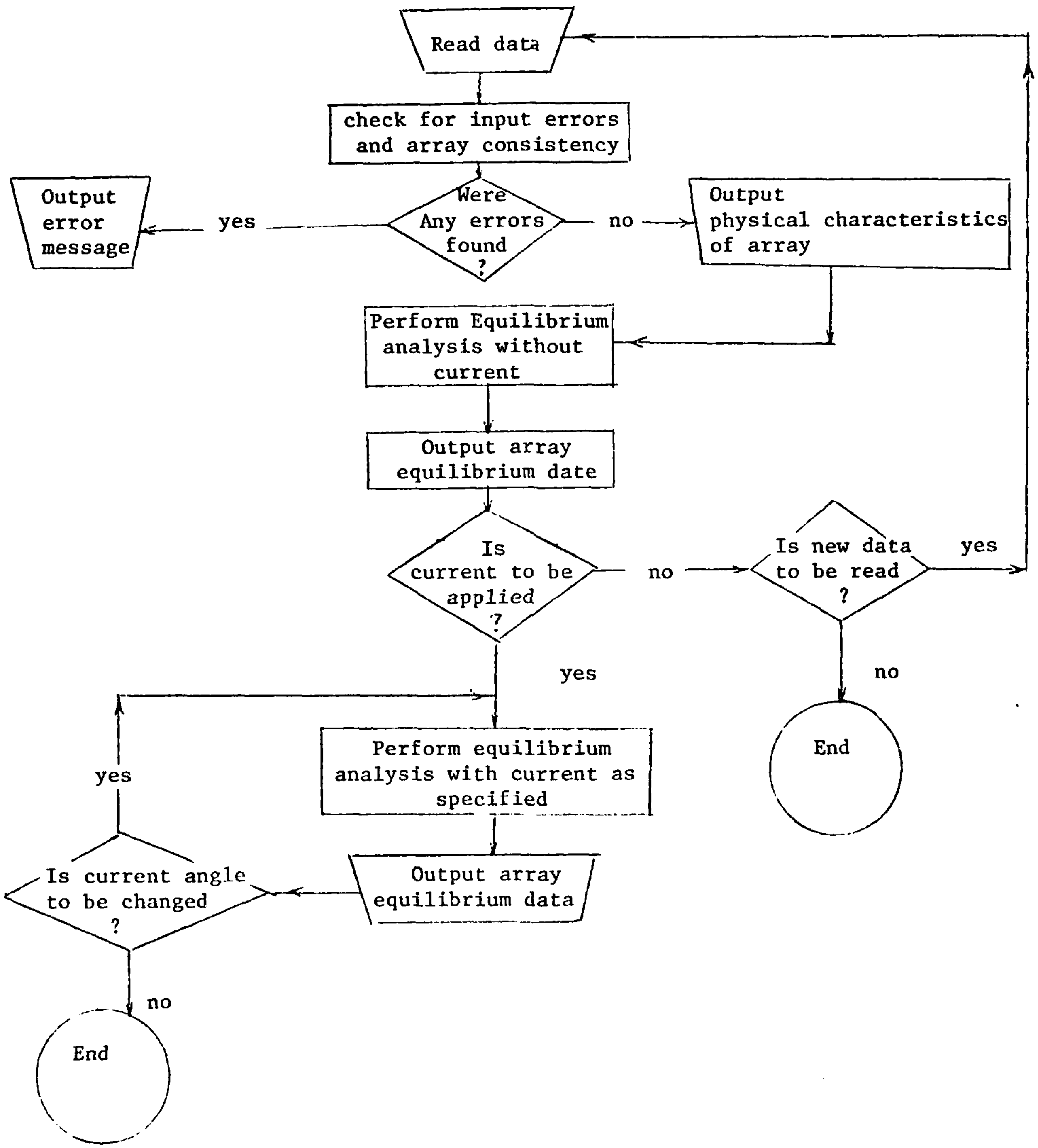
4. Program Structure

In addition to the main program 'DESADE' there are 8 subroutines and 7 functions. These are described below:

Subroutines	Purpose
INPUT	Reads all data and identifies errors in data, deck structure, and array representation.
PHSOUT	Generates information concerning the physical characteristics of the structural cable array.
STROUT	Generates the array structural output.
TAPOUT	Generates the tape or cards giving the locations of the indexed devices.
ERROR	Generates error message when the accuracy required for the array equilibrium calculation could not been obtained.
RPOLY	Finds real roots of polynomial equations for use in evaluating maximum cable displacements and tension extreme.
SWITCH	Switches input data.
START	Calculates the initial guesses.

Functions	Purpose
TCAB(M,K)	Calculates the tension at node M of Cable K.
EXCAB(M,K)	Calculates (1 + strain) at node M of Cable K.
EFORCE (I)	Calculates the hydrodynamic drag force on each device in direction I.
CFORCE (I,M,N)	Calculates the force/length in direction I at node M on Cable N.
SPACE (I)	Calculates the location in space of any point on the array.
TANG (I)	Calculates the unit tangent to a cable at any point.
VELOC (I,P SAPCE)	Specifies the I component of the current field at an arbitrary point in space, PSPACE (I).

5. Program Flow Chart



6. Input Coding

(1) I/O logical unit number card

Column	1 4	8	16	24	32	40	48	56	64
	N_{card}	Card	N_{read}	N_{write}	N_{tape}	$I_{opt.}$	Blank	$M_{opt.}$	$K_{opt.}$

N_{card} = card number, (I4), From - 999 to 9999, used to sequence the data cards

Card = ⌢LUN, (A4), used to catalog each data card

N_{read} = 5, (I8), reader unit number

N_{write} = 6, (I8), printer unit number

N_{tape} = 2, (I8), service tape unit number

$I_{opt.}$ = 0, (I8), Input option using source deck

$M_{opt.}$ = 2 or 0, (I8), Output option to printer

$K_{opt.}$ = 3 or blank, (I8), Output tape unit number

(2) number card of junctions

1 4	8	16
N_{card}	Card	$N_{jun.}$

Card = $N_{jun.}$, (A4), card name

$N_{jun.}$ = number of junction in original Array, (I8)

(See Figure 1), $2 \leq N_{Jun} \leq 44$

(3) Anchor location cards (one for each anchor)

1 4	8	16	24	32	40
N_{card}	Card	N_{jun}	X	Y	Z

Card = ⌢ANC, (A4)

$N_{jun.}$ = Junction number of anchor (original array)

$1 \leq N_{Jun} \leq 44$, (I8)

X = anchor X - coordinate (ft.), (F8.0)

Y = anchor Y - coordinate (ft.), (F8.0)

Z = anchor Z - coordinate (ft.), (F8.0)

Refer to Table 1.

(4) Cut data cards (one for each cut)

1 4	8	16	24
N_{card}	Card	$N_{jun,New}$	$N_{un,Old}$

Card = ⌢⌢IR, (A4)

$N_{Jun,New}$ = New junction number at cut, (I8)

$N_{Jun.Old}$ = Orginial junction number at cut, (I8)

$1 \leq N_{Jun,Old} \leq 44$

Refer to Table 2.

(5) Cable cards (one for each cable)

1 4	8	16	24	32	40	48	56	64	72	77	80
N_{card}	Card	N_c	NJun. s=o	NJun. s=L	W_c	C_{D_c}	d	L	C	h	$N_{ele.}$

Card = ⌢CAB, (A4)

N_c = cable number; (I8), $1 \leq N_c \leq 22$

$N_{Jun,s=o}$ = Junction number at s=o, (I8), $1 \leq N \leq 44$

$N_{Jun,s=L}$ = Junction number at s=L, (I8), $1 \leq N \leq 44$

W_c = cable unit weight in sea water (lb/ft), (F8.0)

(+) if positively buoyant

(+) if negatively buoyant

C_{D_c} = cable normal drag coefficient, (F8.0), (based on cable diameter)

d = cable diameter, (in.), (F8.0)

L = cable unstressed length (ft), (F8.0)

C = cable rigidity (lb), (F8.0)

$(T/C)^k = \varepsilon$, T = tension, ε = Strain, k = constant

k = constant in cable constitutive relation, (F5.0)

Nele. = number of finite elements in cable calculation, (I3),

$0 < N \leq 50$

(6) Device on cable cards (one for each device)

1 4	8	16	24	32	40	48	56	64	72
N_{card}	Card8	N_c	I_{type}	Index	W_d	C_{D_d}	A.	1	S_d

Card = DCAB, (A4)

N_c = number of cable to which device is attached, (I8)

$1 \leq N \leq 22$

I_{type} = device type, (I8), 1 or 3 for in-line elongated device (longitudinal axis aligns with cable axis), 2 or 4 for other free devices or for divices inside the cable.

I_{Index} = Device index if type 1 or 2, (I8) $1 \leq I \leq 1000$, Type 1 and 2 must be indexed consecutively from one to the total number of type 1 and 2 devices in the array.

W_d = device weight in sea water (lb.), (F8.0)

(+) if positively buoyant

(-) if negatively buoyant

C_{D_d} = device drag coefficient, (F8.0)

(based on frontal area or diameter)(blank if device is inside the cable)

A. = Frontal area of device for type 2 or 4 (ft^2), (F8.0)

or diameter of device for type 1 or 3, (in), (F8.0).

l = device length (if type 1 or 3), (ft), (F8.0).

S_d = Unstressed distance of device from s=o junction of the cable, (ft), (F8.0).

(7) Device on junction cards (one for each device)

1 4	8	16	24	32	40	48	56
N_{card}	Card	NJun	I_{type}	I_{dex}	W_d	Co_d	A.

Card = DJNC, (A4)

$N_{Jun.}$ = number of junction to which device is attached, (I8)

$1 \leq N \leq 44$

I_{type} = Device type (2 or 4), (I8)

I_{dex} = Device index if type 2, (I8)

$1 \leq I \leq 1000$ (indexed consecutively from one to the total number of type 1 and 2 devices in the array.

W_d = Device weight in sea water (lb.) (F8.0)

(+) if positively buoyant

(-) if negatively buoyant

C_D = Device drag coefficient based on frontal area (F8.0)

A. = device frontal area (ft^2), (F8.0)

(8) Water density card

1 4	8	16
N_{card}	Card	ρ_f

Card = DEN, (A4)

ρ_f = sea water density, (slug / ft^3), (F8.0)

(9) End of data card

1 4	8
N_{card}	Card

Card =⌃EOD, (A4)

(10) Flag card to specify current data

1 4	8	16
N_{card}	Card	$I_{opt.current}$

Card = NDAT, (A4)

$I_{opt.\ current}$ = Current option = 1 standard

= 2 nonstandard (use modified program)
(I8)

(11) Accuracy card (following the 1st NDAT card)

1 4	8	16
N_{card}	Card	E

Card = COMP, (A4)

$\mathcal{E}$ = accuracy specified (ft), (F8.0)

To insure the calculated coordinates of every point in the array are within $\pm$ E of their exact values. The obtainable $\mathcal{E}$ is limited by the computer capacity and by the largest linear dimension in the array.

$\mathcal{E} \leq 10^{m-n+3}$ where m = common logarithm of the largest linear dimension, n = number of significant figures carried in single precision (e.g. n = 8, L = 25,000 ft, $\mathcal{E} \geq 0.1$ ft.)

$\mathcal{E}$ = 1 ft. was used in the test runs.

(12) Current cards (one for each point)

1 4	8	16	24
N_{card}	Card	z	V(z)

Card = ⌃VEL, (A4)

Z = Z - coordinate of input point, (ft), F8.0)

V(Z) = Velocity at input point, (knots), (F8.0)

Up to 25 VEL cards are permitted. At least on input point must be below or equal the minimum Z coordinate of the anchors.

(13) Current direction card

^{1} N_{card} ^{4}	Card ^{8}	θ_o ^{16}	$\Delta\theta$ ^{24}	θ_f ^{32}

Card = ⌃ANG, (A4)

θ_o = initial current angle (deg.), (F8.0)

$\Delta\theta$ = increment in current angle (deg. >0), (F8.)

θ_f = final current angle (deg. $\geq \theta_o$), (F8.0)

(14) End of data card

^{1} N_{card} ^{4}	Card ^{8}

Card = ⌃ EOD, (A4)

(15) Termination card

^{1} N_{card} ^{4}	Card ^{8}

Card = EOP, (A4)

7. Output Parameters

The outprints are self-explanatory and consist of:

a. Array descriptive output:

- o anchor locations
- o cuts information
- o cable properties
- o device properties
- o current field

b. Structural output:

- o current condition
- o cable forces and angles at each anchor
- o location of the array junctions
- o displacement of these junctions from the no-current coordinates
- o cable forces and angles at each junction
- o maximum and minimum tension and their location for each cable
- o location and displacement of devices from the no-current coordinates
- o tension at devices.

A sample output for the tri-moor is given in Appendix B-2.

8. File Status

NAME	DESCRIPTION	* STATUS ON TAPE	STATUS ON DISK
DESADEK	Source program	X	
DESADW	Source program modified to accept variable current directions.	X	X
DSDLGØK	Compiled program on NCS system	X	
GØDESAD	Procedure file to run by Scope system (Compile, catalog LGØ file, and run.)	X	X
GØDSAD	Procedure file to run by Scope system (run LGØ file.)	X	X
GØDSW	Procedure file to run by Scope system (Compile and run.)	X	X
DATA	Bench data from Ref. 1	X	X
DATASP	Data file for single post near	X	X
DATAGP1	Data file for goal post moor	X	X
DATAGP8	Data file for goal post w/eight additonal buoys on each post.	X	X
DATABGP	Data file for braced goal post post moor.	X	X
DATASS	Data file for sea spider moor.	X	X
DATA4	Data file for single post single buoy moor.	X	X
DATAW	Data file for DOCMS moor:	X	X

*TAPE NAME: KW2233 Created on 8-3-77

9. Computer Requirements and Cost

a. Requirements

DESADE is a Fortran IV program and is ready for CDC Fortran IV compilers. Memory requirements are approximately 30,000 words in a single precision. Access to one, two or three magnetic tape units, depending on the I/O options chosen, is required by the program.

b. Cost

Computer costs for the test runs are shown in the following table :

ARRAY CONFIGURATION	I/O time (SEC)	Execution time* (SEC)	total* SBU UNITS (SEC)	ESTIMATE** TOTAL COST ($)
Single post	14	13	5/30's	9
Sea spider	14	54	110/30's	34
Goal Post	14	24	36/30's	16
Braced Goal Post	14	84	208/10's	58

* CDC Computer charge unit

** Based on over night rate (priority 2) plus terminal connecting charge and I/O charge. Current rates are (July, 1977) :

Pirority	(Overnight)	(Daytime)		
	2	3	4	5
Rate ($)	.24	.28	.34	.45

Terminal connection: $9/hr.
Terminal I/O: $20/1000 character

Note the compilation time (~ 4.5 sec) was saved for each run through the use of a compiled binary file (DESLGO)

10. Run Steps

(1) Log in

```
[illegible] KB,IHJ11*

[illegible]
[illegible]                                    [illegible]
[illegible]
[illegible] ← FP01*
[illegible]
[illegible]
[illegible] CHAR,M437OTG,CABLECOMP*
```

(2) Prepare data file

a. Either create your own file (see next section on how to establish data file).

b. Or edit existing data files of similar array configuration.

c. A sample of data file is shown on page 20 .

(3) Prepare run procedure file

a. Edit existing file to desired condition (job priority, data file name, etc.)

b. A sample of run procedure file is shown on page 19 .

(4) Submit job

a. See example on page 19 .

(5) Check for job status

a. See example on page 19.

(6) Get day file when job is ready for output.

a. See example on page 19.

(7) Print output

a. See example on page 19.

* The underlined characters are user's entry. Consult with FPO-1E to update these entries.

Get Run Procedure File and Edit to suit Need

— priority

```
OLD,GODSAD *                    /JOB
READY.                          CHIH(1),T15.
EDIT *(example)                 USER,[illegible]
 BEGIN TEXT EDITING.            ATTACH,[illegible]
? [illegible]                   LIBSET,[illegible]
? L                             LGO.
/JOB                            REWIND,TAPE[illegible]
? L;2                           COPYSBF,TAPE[illegible],OUTPUT.
/JOB                            /EOR
CHIH,P3,T15?.                   /NOGO[illegible]
? END                           /READ,([illegible])
 END TEXT EDITING.              /EOF
READY.
REPLACE
```

— Data file

Submit Job

```
READY.
SUBMIT,GODSAD,ST=ECZ,T *
 14.42.37 AEOCBQK
```

Assigned job name

Get Day File when Job is Done

```
READY.
QSTATUS*
 FAMILY NOW ACTIVE  KO
 [illegible]
 [illegible]
 [illegible]
 [illegible]
 YOUR LOGIN FAMILY  KO

 FILENAME       [illegible]       [illegible]       STATUS

 1. AEOCBQK     [illegible]       [illegible]       READY FOR [illegible]
READY.
QFETCH,AEOCBQK,DAY*
AEOCBQK RELEASED [illegible]
 [illegible]
 [illegible]
 [illegible]
 [illegible]
 [illegible]
```

— Check for job status

Job AEOCBQK is ready for print.

— Get day file

⋮

Print Output

```
READY.
LNH,F=FILE *
```

* The underlined characters are user's entries.

References

1. R. A. Skop, and J. Mark, 'A Fortran IV program for computing the static deflections of structural cable arrays', NRL Report 7640, August 1973.

2. N. D. Albertsen, 'A survey of techniques for the analysis and design of submerged mooring systems', CEL Report 815, Aug. 1974.

3. T. R. Kretschmer, G.A. Edgerton, and N. D. Albertsen, 'Seafloor construction experiment, SEACON II - an instrumented Tri-moor for evaluating undersea cable structure tehcnology', CEL Report R848, Dec. 1976.

4. R. A. Skop, and G. J. O'Hara, 'The static equilibrium configuration of cable arrays by use of the method of imaginary reactions', NRL report 6819, Feb. 1969.

5. R. A. Skop, and R. E. Kaplan, 'The static configuration of a tri-moored, subsurface, buoy - cable array acted on by current - induced forces', NRL Report 6894, May 1969.

6. R. L. Webster, 'DSSM computer program', GE Report 1976.

APPENDIX A

Results of Test Runs

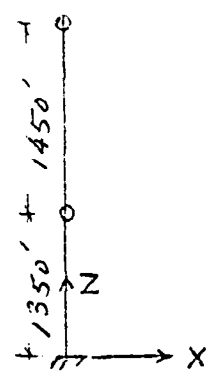
The test runs were made for the Linear Chair candidate configurations. The results have been used to compare the calculations made by DTNSRDC using the DSSM computer program (Ref. 6). In the following tables the DTNSRDC's results are enclosed by prentheses. Good agreement exists in most of the cases. The computer costs are comparable. However the finite element representation for each cable member is much finer in the test run calculations than those made by DTNSRDC.

Single Point Moor — Deviation in feet from zero position

Cable —— $1\frac{1}{4}''$ Dia.

Current —— Uniform = .2 Knots

Buoyancy (lbs)	Top Buoy		Bottom Buoy	
	X/Y - Axis	Z - Axis	X/Y - Axis	Z - Axis
3000	(13.74) 14.6	(5.79) 5.9	(7.94) 8.4	(3.73) 3.8
4000	(10.18) 10.9	(7.87) 8.1	(5.87) 6.3	(5.09) 5.2

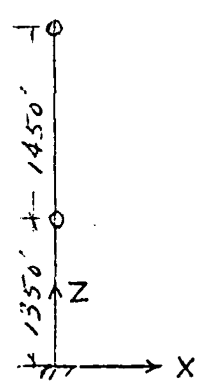


Single Point Moor — Deviation in feet from zero position

Cable ——— 1 1/4" Dia.

Current ——— St. Croix

Buoyancy (lbs)	Top Buoy		Bottom Buoy	
	X/Y - Axis	Z - Axis	X/Y - Axis	Z - Axis
3000	(13.26) 15.5	(5.79) 5.9	(6.03) 6.9	(3.74) 3.8
4000	(10.01) 12.1	(7.88) 8.1	(4.53) 5.3	(5.10) 5.2

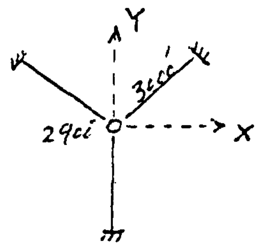


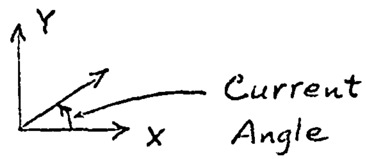
Sea Spider Moor —— Deviation in feet from zero position

Cable —— 1 1/4" Dia. Buoyancy —— 3000 lbs

Current —— Uniform = .2 Knots

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	60°	90°	0°	30°	60°	90°	0°	30°	60°	90°
Top	(1.91) 2.4	(.96) 1.1	(.28) .2	(0) 0	(-.78) -.7	(.55) .7	(2.04) 2.3	(2.71) 3.1	(-5.87) -6.0	(-5.84) -5.9	(-5.87) -6.0	(-5.9) -6.0



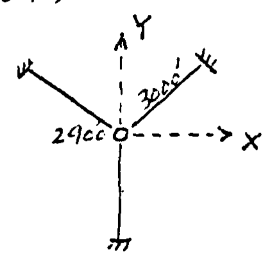


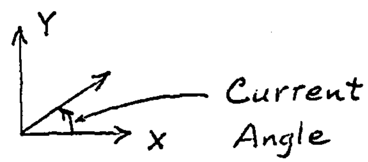
Sea Spider Moor —— Deviation in feet from zero position

Cable —— 1 1/4" Dia. Buoyancy —— 3000 lbs.

Current —— St. Croix

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	60°	90°	0°	30°	60°	90°	0°	30°	60°	90°
Top $(C_D=.12)$	8.7	8.2	5.0		1.0	4.9	7.0		-6.1	-6.2	-6.1	
Top $(C_D=.47)$	10.0 (10.97)	9.7	5.9	0 (0)	1.3 (1.48)	5.7	8.0	8.6 (9.38)	-6.3 (-6.2)	-6.3	-6.2	-6.1 (-6.1)



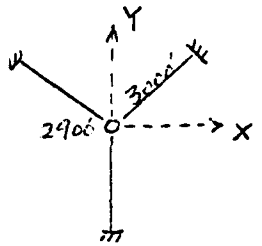


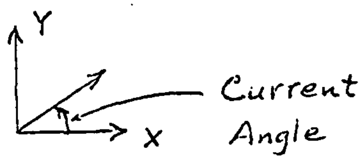
Sea Spider Moor — Deviation in feet from zero position

Cable — $1\frac{1}{4}''$ Dia. Buoyancy — 4000 lbs.

Current — Uniform = .2 knots

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	60°	90°	0°	30°	60°	90°	0°	30°	60°	90°
Top $_{(C_D=.47)}$	(-.84) -.5	(-1.19) -1.4	(-.88) -1.1	(0) 0	(-.53) -.5	(-.69) -.6	(-.46) -.4	(-.31) -.1	(-1.24) -1.3	(-1.23) -1.2	(-1.24) -1.2	(-1.25) -1.3
Top $_{(C_D=.12)}$	-.6	-1.5	-1.2		-.5	-.7	-.4		-1.3	-1.2	-1.2	



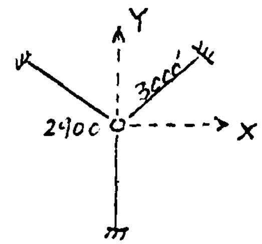


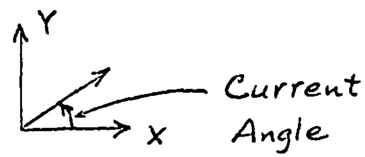
Sea Spider Moor — Deviation in feet from zero position

Cable — 1 1/4" Dia. Buoyancy — 4000 lbs

Current — St. Croix

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	60°	90°	0°	30°	60°	90°	0°	30°	60°	90°
Top	(3.0)			(0) 0	(.19)			(2.87) 2.7	(-1.26)			(-1.25) -1.3



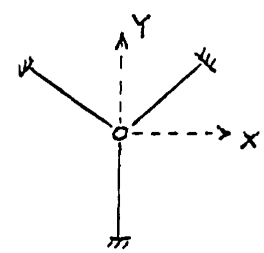


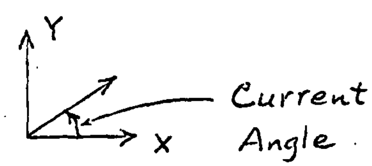
Sea Spider Moor — Deviation in feet from zero position

Cable — $1\frac{1}{4}$" Buoyancy — 3000 lbs

Current — Uniform = .3 kt.

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	60°	90°	0°	30°	60°	90°	0°	30°	60°	90°
Top	(12.0) 13.9	(7.0) 10.2	6.2	(0)	(-1.4) -.7	(4.0) 5.9	12.4		-8.8	-8.4	-8.8	



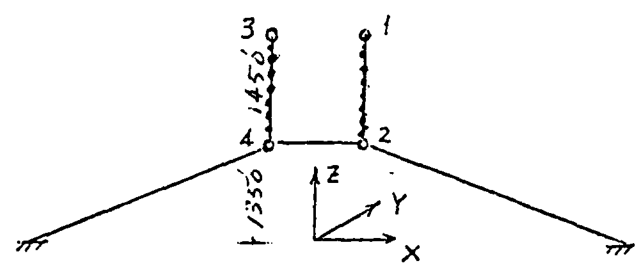


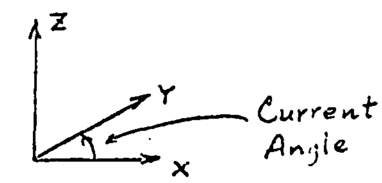
Goal Post Moor — Deviation in feet from zero position

Cable — $1\frac{3}{4}$" Dia. Buoyancy — 2000 lbs plus 8 @ 250 lbs

Current — Uniform = .2 Knots

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°	30°	45°	90°	0°	30°	45°	90°
Top Right 1	(8.72) 8.9		6.4	(.46) .5	0		18.1	(28.27) 28.4	(16.37) 16.5		16.5	(15.75) 16.0
Bottom Right 2	(.66) .7		.6	(.46) .5	0		12.4	(20.21) 20.2	(15.88) 16.0		16.0	(15.26) 15.5
Top Left 3	(7.8) 7.9		5.5	(-.46) .5	0		18.1	(28.27) 28.4	(15.43) 15.6		15.7	(15.75) 16.0
Bottom Left 4	(-.26) -.2		-.3	(-.46) -.5	0		12.4	(20.21) 20.2	(14.94) 15.1		15.2	(15.26) 15.5



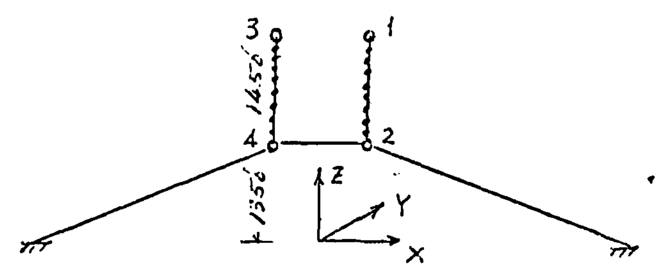


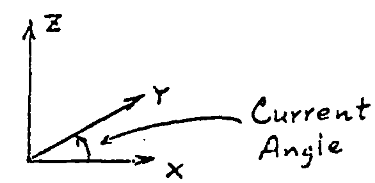
Goal Post Moor — Deviation in feet from zero position

Cable — $1\frac{3}{4}$" Dia. Buoyancy — 2000 lbs plus 8 @ 250 lbs

Current — St. Croix

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°	30°	45°	90°	0°	30°	45°	90°
Top Right 1	(10.7) 11.9		8.5	(.46) .4	0		16.4	(22.15) 24.2	(16.24) 16.4		16.4	(15.85) 16.1
Bottom Right 2	(.62) .7		.6	(.46) .4	0		8.5	(12.09) 13.0	(15.57) 15.9		15.9	(15.37) 15.6
Top Left 3	(9.75) .9		7.6	(-.46) -.5	0		16.4	(22.15) 24.5	(15.57) 15.7		15.8	(15.85) 16.1
Bottom Left 4	(-.3) -.3		-.4	(-.46) -.5	0		8.5	(12.09) 13.0	(15.09) 15.2		15.4	(15.37) 15.6



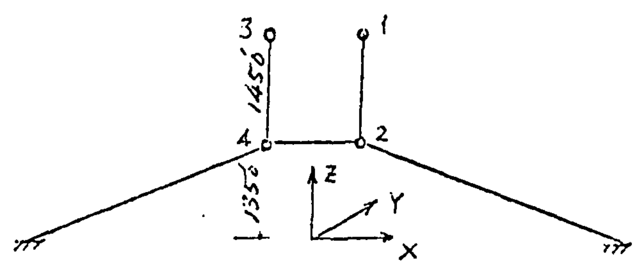


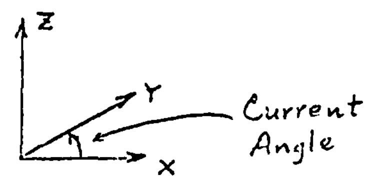
Goal Post Moor — Deviation in feet from zero position

Cable — 1 3/4" Dia. Buoyancy — 3000 lbs.

Current — Uniform = .2 Knots

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°	30°	45°	90°	0°	30°	45°	90°
Top Right 1	(8.69) 9.2		6.6	(.46) .5	(0) 0		18.1	(27.45) 28.4	(16.34) 16.5		16.5	(15.76) 16.0
Bottom Right 2	(.65) .7		.6	(.46) .5	(0) 0		12.1	(19.40) 19.8	(15.85) 16.0		16.0	(15.27) 15.5
Top Left 3	(7.77) 8.3		5.7	(-.46) -.5	(0) 0		18.1	(27.45) 28.4	(15.48) 15.6		15.7	(15.76) 16.0
Bottom Left 4	(-.27) -.2		-.3	(-.46) -.5	(0) 0		12.1	(19.40) 19.9	(14.99) 15.1		15.2	(15.27) 15.5



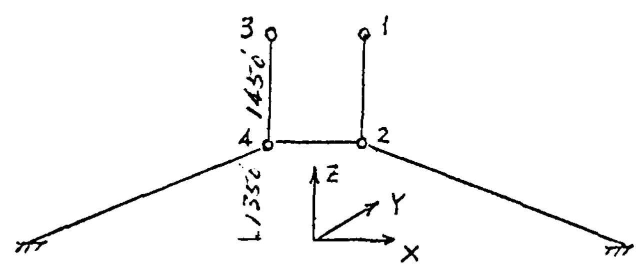


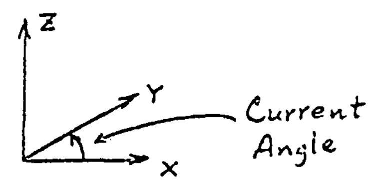
Goal Post Moor — Deviation in feet from zero position

Cable — 1 3/4" Dia. Buoyancy — 3000 lbs.

Current — St. Croix

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°	30°	45°	90°	0°	30°	45°	90°
Top Right 1	(10.85) 12.5		8.9	(.46) .4	(0) 0		16.8	(22.06) 24.9	(16.22) 16.4		16.4	(15.85) 16.1
Bottom Right 2	(.61) .7		.6	(.46) .4	(0) 0		8.5	(11.83) 13.1	(15.75) 15.9		15.9	(15.37) 15.6
Top Left 3	(9.41) 11.5		8.0	(-.46) -.5	(0) 0		16.8	(22.06) 24.9	(15.57) 15.7		15.8	(15.85) 16.1
Bottom Left 4	(-.31) .3		-.4	(-.46) -.5	(0) 0		8.5	(11.83) 13.1	(15.10) 15.2		15.4	(15.37) 15.6



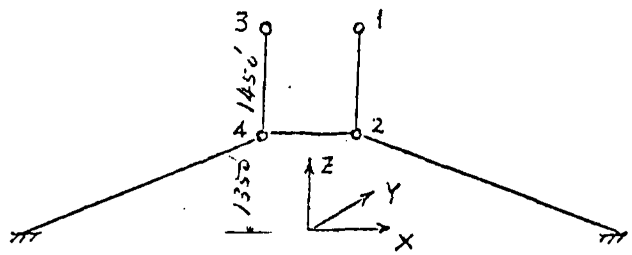


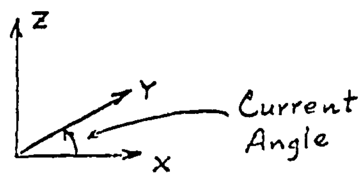
Goal Post Moor — Deviation in feet from zero position

Cable — 1 3/4" Dia. Buoyancy — 4000 lbs

Current — Uniform = .2 Knots

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°	30°	45°	90°	0°	30°	45°	90°
Top Right 1	(6.74) 7.2		5.3	(.64) .6	0		13.3	(20.09) 20.9	(22.90) 23.1		23.3	(22.51) 22.9
Bottom Right 2	(.78) .8		.8	(.64) .6	0		8.8	(14.13) 14.5	(22.21) 22.4		22.6	(21.82) 22.2
Top Left 3	(5.46) 5.9		4.0	(-.64) -.7	0		13.3	(20.09) 20.9	(22.27) 22.5		22.7	(22.51) 22.9
Bottom Left 4	(-.5) -.4		-.5	(-.64) -.7	0		8.8	(14.35) 14.5	(21.59) 21.8		22.0	(21.82) 22.2



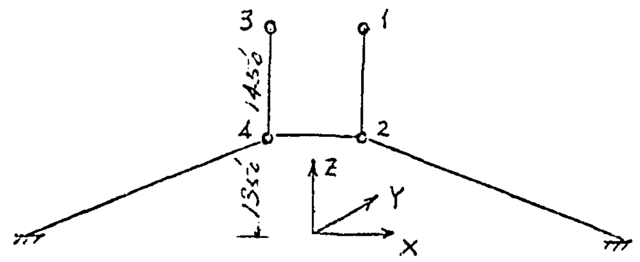


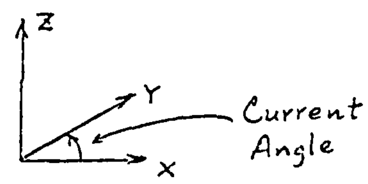
Goal Post Moor — Deviation in feet from zero position

Cable — $1\frac{3}{4}$" Dia. Buoyancy — 4000 lbs.

Current — St. Croix

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45	90°	0°	30°	45°	90°	0°	30°	45°	90°
Top Right 1	(8.61) 10.3		1.4	(.64) .6	(0) 0		13.2	(16.64) 19.3	23.1		23.1	(22.55) 22.9
Bottom Right 2	(.76) .8		.7	(.64) .6	(0) 0		6.5	(8.80) 9.9	22.4		22.5	(21.87) 22.2
Top Left 3	(7.32) 8.9		6.1	(-.64) -.7	(0) 0		13.1	(16.64) 19.3	22.5		22.7	(22.55) 22.9
Bottom Left 4	(-.52) -.5		-.6	(-.64) -.7	(0) 0		6.5	(8.8) 9.9	21.9		22.	(21.87) 22.2



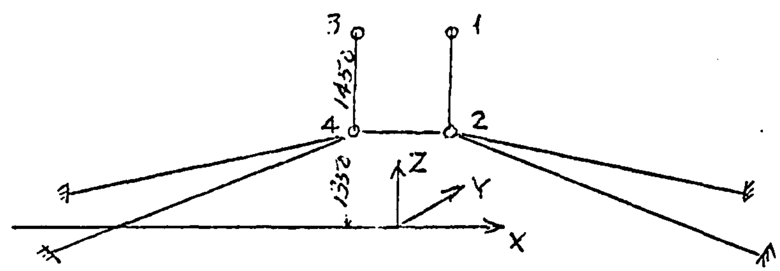


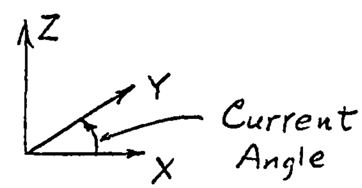
Braced Goal Post Moor — Deviation in feet from zero position

Cable — $1\frac{3}{4}$" Dia. Buoyancy — 3000 lbs.

Current — Uniform = .2 Knots

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	60°	90°	0°		60°	90°	0°		60°	90°
Top Right 1	(8.41) 9.4		5.0	(.34) .4	.2		7.7	(8.24) 8.9	4.3		3.9	(3.15) 3.2
Bottom Right 2	(.86) .9		.7	(.34) .4	.2		.3	(.19) .4	3.8		3.4	(2.67) 2.7
Top Left 3	(8.23) 8.7		4.3	(-.34) -.3	-.1		7.5	(8.24) 8.7	2.1		2.5	(3.15) 3.2
Bottom Left 4	(.19) .2		0	(-.34) -.3	-.1		.2	(.19) .1	1.6		2.0	(2.67) 2.7



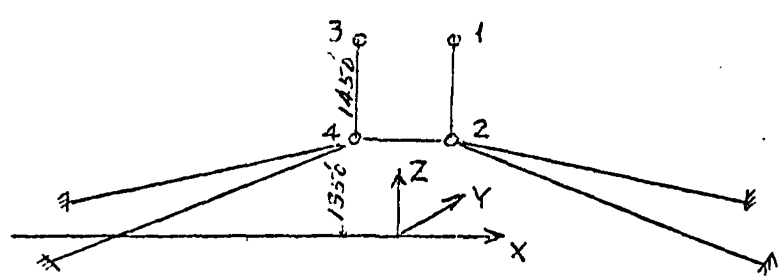


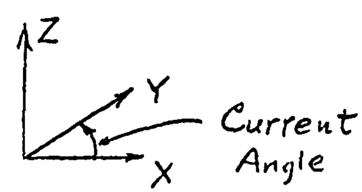
Braced Goal Post Moor — Deviation in feet from zero position

Cable —— 1 3/4" Dia. Buoyancy —— 4000 lbs.

Current —— Uniform = .2 Knots

Buoy Location	X - Axis				Y - Axis				Z - Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°		45°	90°	0°		45°	90°
Top Right 1	(6.82) 7.3		5.3	(.48) .5	(0) .2		4.8	(6.1) 6.7	(7.45) 7.6		7.5	(6.68) 6.8
Bottom Right 2	(.87) .9		.8	(.48) .5	(0) .2		.3	(.15) .3	(6.77) 6.9		6.8	(6.0) 6.1
Top Left 3	(5.86) 6.3		4.4	(-.48) -.5	(0) -.1		4.6	(6.1) 6.5	(5.94) 6.0		6.2	(6.68) 6.8
Bottom Left 4	(-.09) -.1		-.1	(-.48) -.5	(0) -.1		.1	(.15) .1	(5.23) 5.3		5.5	(6.0) 6.1



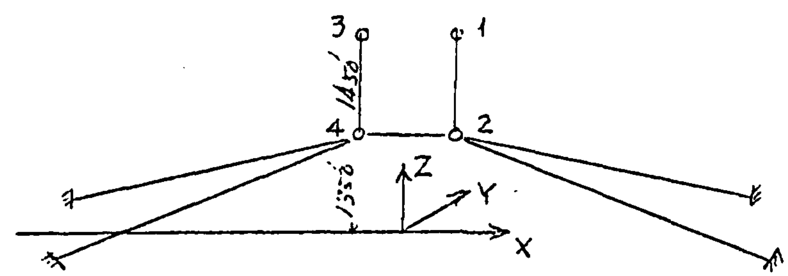


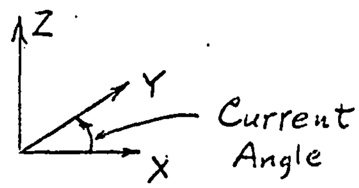
Braced Goal Post Moor — Deviation in feet from zero position

Cable —— $1\frac{3}{4}$" Dia. Buoyancy —— 3000 lbs

Current —— St. Croix

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°		45°	90°	0°		45°	90°
Top Right 1	(11.49) 11.9		6.8	(.34) .4	(0) .2		6.4	(10.9) 11.4	(3.84) 3.8		4.1	(3.16) 3.2
Bottom Right 2	(.7) .7		.8	(.34) .4	(0) .2		.3	(.15) .3	(3.37) 3.4		3.6	(2.69) 2.7
Top Left 3	(10.75) 11.1		6.1	(-.34) -.3	(0) -.1		6.2	(10.9) 11.2	(2.49) 2.4		2.3	(3.16) 3.2
Bottom Left 4	(.05) .1		.1	(-.34) -.3	(0) -.1		.1	(.15) .1	(2.02) 1.9		1.8	(2.69) 2.7



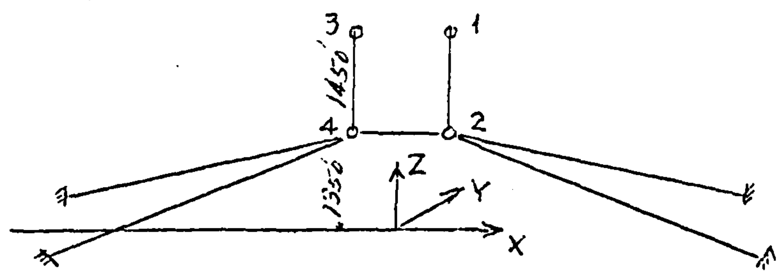


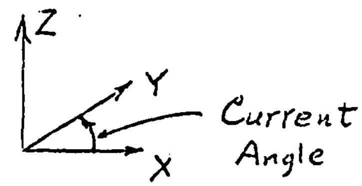
Braced Goal Post Moor — Deviation in feet from zero position

Cable — $1\frac{3}{4}''$ Dia. Buoyancy — 4000 lbs.

Current — St. Croix

Buoy Location	X-Axis				Y-Axis				Z-Axis			
	Current Angle				Current Angle				Current Angle			
	0°	30°	45°	90°	0°		45°	90°	0°		45°	90°
Top Right 1	(8.42) 9.6		6.9	(.48) .5	(0) .2		6.4	(8.76) 9.0	(7.18) 7.3		7.2	(6.69) 6.8
Bottom Right 2	(.74) .8		.7	(.48) .5	(0) .2		.2	(.1) .3	(6.5) 6.6		6.5	(6.01) 6.1
Top Left 3	(7.94) 8.5		5.9	(-.48) -.5	(0) -.1		6.2	(8.26) 8.8	(6.2) 6.3		6.4	(6.69) 6.8
Bottom Left 4	(-.21) -.2		-.2	(.48) .5	(0) -.1		0	(.1) .1	(5.52) 5.6		5.7	(6.01) 6.1





APPENDIX B

Sample of Input and Output

Appendix B-1

Sample Input
(Tri-moor)

INLINE DEVICES ALONG ARRAY CABLES

DEVICE INDEX	CABLE NO.	COORDINATE	TENSION AT DEVICE	DEVICE LOCATION X COORD	Y COORD	Z COORD	DISPLACEMENT FROM NO CURRENT LOC. X DISP	Y DISP	Z DISP

ARRAY LOST IN [illegible]

ARRAY ANCHORS

ANCH. NO. OF ANCHOR	CABLE AT ANCHOR	TENSION AT ANCHOR	FORCE COMPONENTS AT ANCHOR X-COMP	Y COMP	Z COMP	HOR. COMP	CABLE ANGLES WRT X-AXIS	XY-PLANE
1	1	720.6	235.3	120.5	[illegible]	264.6	[illegible]	[illegible]
2	2	[illegible]	4.4	[illegible]	992.1	317.4	[illegible]	78.41
3	3	[illegible]	-187.0	[illegible]	374.6	216.3	[illegible]	59.42

ARRAY CABLES

CABLE NO.	[illegible] TENSION	[illegible]	[illegible] TENSION	S-COORD OF	[illegible] DISP.	S COORD OF	[illegible] X COORD	Y COORD	Z COORD	NO CURRENT LOC. OF [illegible] X COORD	Y-COORD	Z-COORD
1	[illegible]	[illegible]	780.6	3000.0	9.9	1320.0	243.8	152.1	[illegible]	[illegible]	143.4	[illegible]
2	[illegible]	3000.0	945.9	0.0	4.2	1060.0	-.1	[illegible]	[illegible]	0.0	[illegible]	1794.2
3	1193.[illegible]	0.0	612.8	3000.0	5.8	1500.0	[illegible]	170.0	1434.8	266.9	[illegible]	[illegible]

ARRAY JUNCTIONS

JUNC. NO.	CABLE AT JUNCTION	TENSION AT JUNCTION	CABLE-ANGLES WRT X-AXIS	XY-PLANE	JUNCTION LOCATION X-COORD	Y COORD	Z COORD	DISPLACEMENT FROM NO CURRENT LOC. X-DISP	Y DISP	Z-DISP
4	1	[illegible]	144.57	-70.65	.0	1.7	2029.0	.0	2.0	-.1
4	2	[illegible]	-87.36	-79.64						
4	3	1193.2	32.94	-70.20						

INLINE DEVICES ALONG ARRAY CABLES

DEVICE INDEX	CABLE NO.	S COORDINATE	TENSION AT DEVICE	DEVICE LOCATION X-COORD	Y-COORD	Z-COORD	DISPLACEMENT FROM NO CURRENT LOC. X-DISP	Y DISP	Z-DISP

ANALYSIS COMPLETED

ARRAY JUNCTIONS

JUNC. NO.	CABLE AT JUNCTION	TENSION AT JUNCTION	CABLE ANGLES WRT X-AXIS	CABLE ANGLES WRT XY-PLANE	JUNCTION LOCATION X-COORD	JUNCTION LOCATION Y COORD	JUNCTION LOCATION Z COORD	DISPLACEMENT FROM NO CURRENT LOC. X DISP	Y DISP	Z DISP
4	1	[illegible]	147.59	-77.04	2.4	.0	2070.0	2.4	[illegible]	.1
4	2	[illegible]	-84.61	-79.65						
4	3	[illegible]	27.07	-70.23						

INDEXED DEVICES ALONG ARRAY CABLES

DEVICE INDEX	CABLE NO.	S COORDINATE	TENSION AT DEVICE	DEVICE LOCATION X-COORD	Y-COORD	Z-COORD	DISPLACEMENT FROM NO CURRENT LOC. X-DISP	Y-DISP	Z-DISP

1ARRAY EQUILIBRIUM WITH CURRENT FROM 30.00 DEGREES

ARRAY ANCHORS

JUNC. NO. OF ANCHOR	CABLE AT ANCHOR	TENSION AT ANCHOR	FORCE COMPONENTS AT ANCHOR X-COMP	Y-COMP	Z-COMP	HOR. COMP	CABLE ANGLES WRT X-AXIS	XY-PLANE
1	1	876.7	260.2	-141.2	825.2	296.0	-28.40	70.27
2	2	876.2	7.8	295.9	824.7	296.0	88.49	70.22
3	3	508.2	-180.2	104.0	550.2	208.0	150.01	67.29

ARRAY CABLES

CABLE NO.	MAXIMUM TENSION	S-COORD OF	MINIMUM TENSION	S-COORD OF	MAXIMUM DISP.	S COORD OF	LOCATION OF THIS POINT X-COORD	Y-COORD	Z-COORD	NO CURRENT LOC. OF THIS POINT X-COORD	Y COORD	Z-COORD
1	1456.1	0.0	876.7	3000.0	7.9	1320.0	-244.7	150.4	1609.8	-240.6	143.4	1609.8
2	1455.7	3000.0	876.2	0.0	7.8	1680.0	7.8	-286.9	1607.8	0.0	287.2	1607.7
3	1167.7	0.0	508.2	3000.0	3.8	2280.0	465.7	268.9	682.3	469.0	270.7	683.7

ARRAY JUNCTIONS

JUNC. NO.	CABLE AT JUNCTION	TENSION AT JUNCTION	CABLE ANGLES WRT X-AXIS	CABLE ANGLES WRT XY-PLANE	JUNCTION LOCATION X-COORD	JUNCTION LOCATION Y COORD	JUNCTION LOCATION Z COORD	DISPLACEMENT FROM NO CURRENT LOC. X-DISP	Y DISP	Z DISP
4	1	1456.1	145.50	-78.89	1.6	1.1	2070.0	1.6	1.2	.1
4	2	1455.7	-85.57	-78.89						
4	3	1167.7	30.00	-78.16						

ARRAY CABLES

CABLE NO.	MAXIMUM TENSION	S-COORD OF	MINIMUM TENSION	S-COORD OF	MAXIMUM DISP.	S-COORD OF	LOCATION OF THIS POINT X COORD	Y COORD	Z COORD	NO CURRENT LOC. OF THIS POINT X COORD	Y COORD	Z COORD
1	1359.5	0.0	700.0	3000.0								
2	1360.1	3000.0	780.7	0.0								
3	1359.5	0.0	700.0	3000.0								

JUNC. NO.	CABLE AT JUNCTION	TENSION AT JUNCTION	CABLE ANGLES WRT X-AXIS	XY-PLANE	JUNCTION LOCATION X COORD	Y COORD	Z COORD	DISPLACEMENT FROM NO CURRENT LOC. X DISP	Y DISP	Z DISP
4	1	1359.5	149.99	-70.70	0.0	.1	2090.9			
4	2	1360.1	-90.00	-70.70						
4	3	1359.5	30.01	-70.70						

INDEXED DEVICES ALONG ARRAY CABLES

DEVICE INDEX	CABLE NO.	S COORDINATE	TENSION AT DEVICE	DEVICE LOCATION X COORD	Y COORD	Z COORD	DISPLACEMENT FROM NO CURRENT LOC. X DISP	Y DISP	Z DISP

1ARRAY EQUILIBRIUM WITH CURRENT FROM 0.00 DEGREES

ARRAY ANCHORS

JUNC. NO. OF ANCHOR	CABLE AT ANCHOR	TENSION AT ANCHOR	FORCE COMPONENTS AT ANCHOR X-COMP	Y-COMP	Z COMP	HOR. COMP	CABLE ANGLES WRT X AXIS	XY PLANE
1	1	946.2	277.1	154.7	891.4	317.5	29.16	70.40
2	2	700.1	9.1	266.4	753.2	266.6	88.04	70.02
3	3	614.9	-104.9	-111.7	575.7	216.0	140.86	69.51

ARRAY CABLES

CABLE NO.	MAXIMUM TENSION	S COORD OF	MINIMUM TENSION	S-COORD OF	MAXIMUM DISP.	S COORD OF	LOCATION OF THIS POINT X COORD	Y COORD	Z COORD	NO CURRENT LOC. OF THIS POINT X COORD	Y COORD	Z COORD
1	1325.6	0.0	946.2	3000.0	4.5	1080.0	196.3	118.3	1043.1	199.3	115.0	[illegible]
2	1359.6	3000.0	700.1	0.0	10.0	1600.0	10.0	207.1	1609.7	0.0	207.2	[illegible]
3	1194.4	0.0	614.9	3000.0	5.7	1440.0	276.4	153.0	1472.7	274.0	150.1	[illegible]

5 1
6 3

NO. OF CABLES IS 3

CABLE NO.	S=0 JUNC	S=L JUNC	LENGTH	DIAMETER	WEIGHT/LENGTH	DRAG COEFFICIENT	RIGIDITY	CONSTITUTIVE EXPONENT	NO. OF ELEMENTS
1	4	5	3000.0	1.250	-.200	1.270	1963500.	1.000	50
2	2	4	3000.0	1.250	-.200	1.270	1963500.	1.000	50
3	4	6	3000.0	1.250	-.200	1.270	1963500.	1.000	50

PROPERTIES OF THE DEVICES LOCATED AT JUNCTIONS ARE AS FOLLOWS

DEVICE JUNC. NO.	DEVICE WEIGHT	DEVICE DRAG COEFFICIENT	DEVICE FRONTAL AREA
4	4000.00	.120	20.50

TOTAL NO. OF INDEXED DEVICES IS 1

CURRENT FIELD OPTION IS 1

Z COORDINATE OF CURRENT	VELOCITY OF CURRENT AT Z
0.00	.10
400.00	.10
2400.00	.10
2700.00	.24
2900.00	.53

ACCURACY REQUIRED IN CALCULATIONS IS 1.00

DEVICE LOCATION OUTPUT RECORD 1 REFERS TO THIS ARRAY
1ARRAY EQUILIBRIUM WITH NO CURRENT

ARRAY ANCHORS

JUNC. NO. OF ANCHOR	CABLE AT ANCHOR	TENSION AT ANCHOR	X-COMP	Y-COMP	Z COMP	HOR.-COMP	CABLE ANGLES WRT X AXIS	CABLE ANGLES WRT XY PLANE
1	1	730.0	230.7	133.3	723.1	266.4	30.01	[illegible]
2	2	730.7	0.0	266.5	723.0	266.5	90.00	[illegible]

B-4

```
        FWA OF THE LOAD           101
        LWA+1 OF THE LOAD      110713

        TRANSFER ADDRESS -- DESADE        65502

               .403 CP SECONDS          124100B CM STORAGE USED                 65 TABLE MOVES
 1
-1ERRORS IN PARAMETRIC STUDY SOURCE DECK NUMBER  0

            CARD   CARD      OTHER
      TYPE   NO.   TYPE   INFORMATION


      NO ERRORS DETECTED
 1ERRORS IN PARAMETRIC STUDY SOURCE DECK NUMBER  1

            CARD   CARD      OTHER
      TYPE   NO.   TYPE   INFORMATION


      NO ERRORS DETECTED
 1PHYSICAL CHARACTERISTICS OF THE STRUCTURAL CABLE ARRAY


    NO. OF ANCHORS IS  3

        JUNCTION NO.    X-COORDINATE       Y-COORDINATE       Z-COORDINATE
             1             -665.20            384.10              0.00
             2                0.00           -768.10              0.00
             3              665.20            384.10              0.00

    NO. OF JUNCTIONS IN ORIGINAL ARRAY IS  4

    NO. OF CUTS MADE IN ORIGINAL ARRAY IS  2

         JUNCTION NO.     JUNCTION NO.
           OF CUT      AT WHICH CUT MADE
             5                1
             6                3
```

Sample Output
(Tri-moor)

Appendix C

Source Program Listing

```
      PROGRAM DESADE(INPUT,OUTPUT, TAPE2, TAPE3, TAPE5=INPUT, TAPE6=OUTPUT)
C
C A FORTRAN IV PROGRAM FOR COMPUTING THE STATIC DEFLECTIONS
C OF STRUCTURAL CABLE ARRAYS
C
C BY RICHARD A. SKOP AND JAMES MARK, OCEAN TECHNOLOGY DIVISION,
C NAVAL RESEARCH LABORATORY, WASHINGTON, D. C.
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,L,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAJ,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      DIMENSION DATAT(2150,10)
      EQUIVALENCE (DATAT(1),FEJUNC(1))
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
C
C CALL INPUT TO READ DATA AND IDENTIFY ERRORS
C
      ITIME=1
 1000 CALL INPUT
C
C CHECK TO SEE IF ANY ERRORS IN DATA
C
      IF(FATE.NE.0.) GO TO 10000
C
C GET HERE IF NO ERRORS -- PRINT OUT PHYSICAL CHARACTERISTICS OF ARRAY
C
      WRITE(IPRNT,1001)
 1001 FORMAT(///,5X,18HNO ERRORS DETECTED)
      CALL PHSOUT
C
C KMULT IS A MULTIPLIER FOR CHANGING CURRENT ANGLE THETA
C
      KMULT=0
C
C JUMP=0---NO CURRENT   JUMP=1---CURRENT
C
      JUMP=0
C
C GET HERE TO CALCULATE FORCES AND IF SUCCESSIVE APPROXIMATION ROUTINE
C NOT SATISFIED -- ZERO FORCES
```

```
C
 2011 DO 2012 J=1,NJUNC
      DO 2012 I=1,3
 2012 FEJUNC(I,J)=0.
      DO 2013 N=1,NCAB
      INNN=NNODE(N)
      DO 2013 M=1,INNN
      DO 2013 I=1,3
 2013  FCAB(I,M,N)=0.
C
C PICK UP DISCRETE DEVICE DATA FROM TEMPORARY STORAGE TAPE AND
C CALCULATE DEVICE FORCES -- DJNC FORCES ARE STORED IN FEJUNC --
C DCAB FORCES IN FCAB -- EFORCE(I) IS ROUTINE FOR CALCULATING
C DEVICE FORCES IN DIRECTION I
C
 2015 READ(INTAPE,1)  (DATA(K),K=1,10)
    1 FORMAT(F4.0,A4,8E15.8)
      IF(DATA(2).EQ.TEST(3))  GO TO 2017
      IF(DATA(2).EQ.TEST(4))  GO TO 2019
      IF(DATA(2).EQ.TEST(9))  GO TO 2021
      GO TO 2015
 2017 K=DATA(3)
      DO 2018 J=1,3
      I=J
 2018 FEJUNC(I,K)=FEJUNC(I,K)+EFORCE(I)
      GO TO 2015
 2019 K= DATA(3)
      M= DATA(10)/H(K) + 1
      DO 2020 J=1,3
      I=J
 2020  FCAB(I,M,K) =  FCAB(I,M,K) + EFORCE(I)
      GO TO 2015
C
C GET HERE AT END OF TAPE
C
 2021 REWIND INTAPE
C
C NOW CALCULATE THE FORCE/LENGTH IN DIRECTION I AT NODE M ON CABLE N
C CFORCE(I,M,N) IS ROUTINE FOR DOING THIS -- INTEGRATE BY TRAPEZOIDAL
C RULE OVER SEGMENT TO GET TOTAL FORCE AND ADD TO DCAB FORCES
C
      DO 2022 J=1,NCAB
      K=J
      INNN=NNODE(K)-1
      DO 2022 MM=1,INNN
      M=MM
      DO 2022 II=1,3
      I=II
 2022 FCAB(I,M,K)=(CFORCE(I,M,K)+CFORCE(I,M+1,K))*H(K)/2.+FCAB(I,M,K)
C
C ALL FORCES ARE NOW CALCULATED AND EQUILIBRIUM CAN BE DETERMINED
C  LEAP = 1  FIRST TIME THROUGH IMAGINARY REACTION ROUTINE
C  LEAP = 2  ANY OTHER TIME
C SKIP THIS SECTION IF NO IR'S
C
      IF(NIR.EQ.0) GO TO 2031
      IF(JUMP.EQ.0) CALL START
      LEAP=1
C
```

```
C INITIALIZE DELTA
C
      DELTA=DELTAI
C
C GET HERE TO INITIALIZE TOTAL FORCES AT THE JUNCTIONS, TFJUNC, AND
C IF THE IMAGINARY REACTION ITERATION NOT SATISFIED -- ALSO IF NO IR'S
C
 2031 DO 2032 J=1,NJUNC
      DO 2032 I=1,3
 2032 TFJUNC(I,J) = FEJUNC(I,J)
C
C ADD APPROPRIATE REACTIONS TO TFJUNC -- SKIP THIS SECTION IF NO IR'S
C
      IF(NIR.EQ.0) GO TO 2036
      DO 2035 J=1,NJUNC
      DO 2035 K=1,NIR
      IF( (J.EQ.IRJUNC(K)).OR.(J.EQ.ERJUNC(K)) )  GO TO 2033
      GO TO 2035
 2033 DO 2034 I=1,3
 2034 TFJUNC(I,J) = TFJUNC(I,J) + IR(I,J)
 2035 CONTINUE
C
C TFJUNC IS NOW DETERMINED AND THE REACTIVE FORCES IN THE ARRAY, RCAB,
C CAN BE CALCULATED BY SUMMING FROM THE FREE ENDS TO THE FIXED ANCHOR
C
 2036 DO 2047 N=1,NCAB
      INDEX= NCAB+1-N
      K= PATH(INDEX)
      INNN=NNODE(K)
      INLJ=LJUNC(K)
      DO 2041 I=1,3
 2041 RCAB(I,INNN,K)=TFJUNC(I,INLJ)
      DO 2044 L=1,NCAB
      IF(LJUNC(K).EQ.ZJUNC(L)) GO TO 2042
      GO TO 2044
 2042 DO 2043 I=1,3
 2043 RCAB(I,INNN,K)=RCAB(I,INNN,K)+RCAB(I,1,L)
 2044 CONTINUE
      DO 2046 MM=1,INNN
      M=INNN+1-MM
      IF(M.EQ.1) GO TO 2047
      DO 2045 I=1,3
 2045 RCAB(I,M-1,K)=RCAB(I,M,K)+ FCAB(I,M-1,K)
 2046 CONTINUE
 2047 CONTINUE
C
C ALL REACTIVE FORCES ARE NOW DETERMINED AND THE CONFIGURATION OF THE
C ARRAY, PCAB AND PJUNC, CAN BE FOUND BY INTEGRATING FROM THE FIXED
C ANCHOR TO THE FREE ENDS -- INTEGRATION BY THE TRAPEZOIDAL RULE IS
C AGAIN USED
C
      DO 2054 N=1,NCAB
      K=PATH(N)
      INLN=LJUNC(K)
      INNN=NNODE(K)
      INZJ=ZJUNC(K)
      DO 2051  I=1,3
 2051 PCAB(I,1,K)=PJUNC(I,INZJ)
      DO 2052 MM=2,INNN
```

```
      M=MM
      DO 2052 II=1,3
      I=II
 2052 PCAB(I,M,K)=PCAB(I,M-1,K)+(EXCAB(M-1,K)*RCAB(I,M-1,K)/TCAB(M-1,K)
     1+EXCAB(M,K)*RCAB(I,M,K)/TCAB(M,K))*H(K)/2.
      DO 2053 I=1,3
 2053 PJUNC(I,INLN)=PCAB(I,INNN,K)
 2054 CONTINUE
C
C ARRAY CONFIGURATION NOW DETERMINED -- CHECK TO SEE IF IT SATISFIES
C GEOMETRIC CONSTRAINTS -- SKIP THIS SECTION IF NO IR'S
C
      IF(NIR.EQ.0) GO TO 2071
C
C  CALCULATE ERROR E
C
      E2=0.
      DO 2061 N=1,NIR
      KEN=ERJUNC(N)
      KIN=IRJUNC(N)
      DO 2061 I=1,3
 2061 E2=E2+( PJUNC(I,KEN) - PJUNC(I,KIN) )**2
      E=SQRT(E2)
C
C COMPARE ERROR TO ACCURACY REQUIREMENTS
C
      IF(E.LE.COMPD/10.) GO TO 2071
C
C GET HERE IF GEOMETRIC CONSTRAINTS NOT SATISFIED
C
      GO TO (2062,2065),LEAP
C
C GET HERE FIRST TIME THROUGH IMAGINARY REACTION ROUTINE
C
 2062 LEAP=2
C
C STORE SUCCESSFUL POSITIONS AND REACTIONS
C
 2063 ES=E
      DO 2064 N=1,NIR
      KEN=ERJUNC(N)
      KIN=IRJUNC(N)
      DO 2064 I=1,3
      PJUNCS(I,KEN)=PJUNC(I,KEN)
      PJUNCS(I,KIN)=PJUNC(I,KIN)
 2064 IRS(I,KIN)=IR(I,KIN)
      GO TO 2066
C
C GET HERE ANY OTHER TIME THROUGH IMAGINARY REACTION ROUTINE
C SEE IF ITERATION SUCCESSFUL
C
 2065 IF(E.LT.ES) GO TO 2063
C
C REDUCE DELTA IF NOT SUCCESSFUL INTERATION
C
      DELTA=DELTA/2.
C
C CALCULATE NEW IMAGINARY AND EQUILIBRATING REACTIONS AND GO BACK TO
C RECALCULATE ARRAY EQUILIBRIUM
```

```
C
 2066 DO 2067 N=1,NIR
      KEN=ERJUNC(N)
      DO 2067 I=1,3
 2067 IR(I,KEN)=0.
      DO 2068 N=1,NIR
      KEN=ERJUNC(N)
      KIN=IRJUNC(N)
      DO 2068 I=1,3
      IR(I,KIN)=IRS(I,KIN)+DELTA*(PJUNCS(I,KEN)-PJUNCS(I,KIN))/ES
 2068 IR(I,KEN)=IR(I,KEN)-IR(I,KIN)
C
C CHECK CHANGES IN IMAGINARY REACTIONS
C
      DO 2070 N=1,NIR
      KIN=IRJUNC(N)
      DO 2070 I=1,3
      IF(IR(I,KIN).NE.IRS(I,KIN)) GO TO 2031
 2070 CONTINUE
C NO CHANGES -- TIME TO QUIT
C
      CALL ERROR
      GO TO 10000
C
C GET HERE IF ACCURACY REQUIREMENTS SATISFIED OR NO IR'S
C OUTPUT EQUILIBRIUM IF NO CURRENT -- IF CURRENT, FIRST CHECK TO SEE
C IF ACCURACY REQUIREMENT SATISFIED BY SUCCESSIVE APPROXIMATIONS
C
 2071 JUM=JUMP+1
      GO TO (2072,2075),JUM
 2072 IF((OFLG.EQ.0).OR.(OFLG.EQ.2)) CALL STROUT
      IF((OFLG.EQ.1).OR.(OFLG.EQ.2)) CALL TAPOUT
      IF(OFLG.EQ.1) GO TO 2100
      IF(JUM.NE.1) GO TO 2100
      DO 2200 N=1,NCAB
      INNN=NNODE(N)
      DO 2200 M=1,INNN
      DO 2200 I=1,3
      PCABO(I,M,N)=PCAB(I,M,N)
 2200 RCABO(I,M,N)=RCAB(I,M,N)
      DO 2201 N=1,NJUNC
      DO 2201 I=1,3
 2201 PJUNCO(I,N)=PJUNC(I,N)
C
C APPLY CURRENT IF REQUIRED
C
 2100 IF(IVOPT.EQ.0)  GO TO 9999
      JUMP=1
      THETA=THETAB + KMULT*THETAS
      IF(THETA.GT.THETAE) GO TO 9999
      KMULT=KMULT+1
C
C STORE EXISTING CONFIGURATION FOR COMPARISON PURPOSES
C
 2073 DO 2074 N=1,NCAB
      INNN=NNODE(N)
      DO 2074 M=1,INNN
      DO 2074 I=1,3
 2074 PCABE(I,M,N)=PCAB(I,M,N)
```

```
C
C RECALCULATE FORCES
C
      GO TO 2011
C
C GET HERE IF CURRENT -- CHECK SUCCESSIVE APPROXIMATION ACCURACY
C
 2075 DO 2077 N=1,NCAB
      DO 2077 M=1,INNN          INNN = NNODE(N)
      U=0.
      DO 2076 I=1,3
 2076 U=U+(PCABE(I,M,N)-PCAB(I,M,N))**2
C
C IF NOT ACCURATE STORE CONFIGURATION AND RECALCULATE FORCES
C
      IF(SQRT(U).GT.COMPD) GO TO 2073
 2077 CONTINUE
C
C GET HERE IF POSITION ACCURATE AND OUTPUT POSITION
C
      GO TO 2072
C
C GO BACK FOR MORE DATA
C
 9999 GO TO 1000
10000 CONTINUE
      WRITE(IPRNT,10001)
10001 FORMAT(1H1)
      END
```

```
      SUBROUTINE INPUT
C
C THIS ROUTINE READS ALL DATA AND IDENTIFIES ERRORS IN
C DATA, DECK STRUCTURE, AND ARRAY REPRESENTATION
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAD,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      DIMENSION DATAT(2150,10)
      DIMENSION ITEST(14)
      EQUIVALENCE (DATAT(1),FEJUNC(1))
      EQUIVALENCE (ITEST(1),TEST(1))
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
C
C THIS IS THE BEGINNING OF THE INPUT SECTION
C INITIALIZE ALL CONSTANTS, FLAGS, ARRAYS, AND COUNTERS
C
      IF(ITIME.NE.1)  GO TO 999
           ITEST(1)=4H  IR
           ITEST(2)=4H ANC
           ITEST(3)=4HDJNC
           ITEST(4)=4HDCAB
           ITEST(5)=4H CAB
           ITEST(6)=4H DEN
           ITEST(7)=4HCOMP
           ITEST(8)=4H ANG
           ITEST(9)=4H EOD
           ITEST(10)=4HNJNC
           ITEST(11)=4H VEL
           ITEST(12)=4HNDAT
           ITEST(13)=4H LUN
           ITEST(14)=4H EOP
      PIP=3.14159265/180.
      IFEOD=0
      NDATC=0
      KFLG=0
      DO 1000 I=1,1000
 1000 IDEV(I)=0
      DO 1001 I=1,44
 1001 ICHECK(I)=0
      DO 1002 I=1,22
 1002 ICAB(I)=0
      NANC=0
```

```
      IFRHO=0
      IFJNC=0
      NIR=0
      IROW=1
  999 FATE=0.
C
C CHECK LOGICAL UNITS
C
      IF(ITIME.GT.1)  GO TO  1003
      ITIME=2
      READ 1004,(DATA(I),I=1,10)
 1004 FORMAT(F4.0,A4,8F8.0,F5.0,I3)
      IF(DATA(2).NE.TEST(13)) GO TO 9018
      IPRNT=DATA(4)
C
C GENERATE ERROR MESSAGE HEADER
C
 998  WRITE(IPRNT,9100) NDATC
 9100 FORMAT(1H1,46HERRORS IN PARAMETRIC STUDY SOURCE DECK NUMBER ,I2,//
     1 12X,4HCARD,3X,4HCARD,6X,5HOTHER/1H ,4X,4HTYPE,4X,3HNO.,3X,4HTYPE,
     23X,11HINFORMATION )
      IF(IFEOD.EQ.0) GO TO 13
      IF(IFEOD.NE.0) GO TO 12701
C
C READ ONE INPUT RECORD INTO DATA ARRAY
C
 1003 CONTINUE
      IF(IFLG.EQ.0) READ(IREAD,1004) (DATA(I),I=1,10),EX,NSEG
      IF(IFLG.EQ.1) READ(IREAD,1104) (DATA(I),I=1,10),EX,NSEG
 1104 FORMAT(F4.0,A4,8E15.8,/E12.5,I3)
C
C TYPE AND BRANCH
C
      IF(IFEOD.EQ.0)  GO TO 1006
      IF((DATA(2).EQ.TEST(12)).OR.(DATA(2).EQ.TEST(14)))  GO TO 1006
      GO TO 9018
 1006 DO 1005 I=1,14
      IF(DATA(2).EQ.TEST(I)) GO TO (1,2,3,4,5,6,9018,9018,9,10,9018,
     1 12,9018,14),I
 1005 CONTINUE
C
C GET HERE IF CARD UNIDENTIFIABLE
C
      GO TO 9000
C
C GET HERE IF IR CARD READ
C
    1 IF(DATA(3)-DATA(4))  101,9001,101
  101 DO 102 I=3,4
      IF( (DATA(I).LT. 1.).OR.(DATA(I).GT.44.))  GO TO 9001
  102 CONTINUE
C
C GET HERE IF DATA OK
C COUNT IR AND STORE DATA
C
      NIR=NIR+1
      IRJUNC(NIR)=DATA(3)
      ERJUNC(NIR)=DATA(4)
      DO 103 N=1,NIR
```

```
      IF(IRJUNC(NIR).EQ.ERJUNC(N))  GO TO 9004
      IF( (IRJUNC(NIR).EQ.IRJUNC(N) ) .AND. (NIR.NE.N) )    GO TO 9004
      IF( (ERJUNC(NIR).EQ.IRJUNC(N))) GO TO 9004
  103 CONTINUE
      GO TO 8000
C
C GET HERE IF ANC CARD READ
C
    2 INDEX=DATA(3)
      IF((DATA(3).LT.1.).OR.(DATA(3).GT.44.))  GO TO 9001
      IF(ICHECK(INDEX).NE.0) GO TO 9002
C
C GET HERE IF DATA OK -- COUNT ANCHOR AND STORE DATA
C
      NANC=NANC+1
      ANJUNC(NANC)=DATA(3)
      ICHECK(INDEX)=1
   22 DO 21 I=1,3
   21 PJUNC(I,INDEX)=DATA(I+3)
      GO TO 8000
C
C GET HERE IF DJNC CARD READ
C
    3 IF( (DATA(4).EQ.1.).OR.(DATA(4).EQ.3.) ) GO TO 9001
      IF( (DATA(4).EQ.2.).AND.(DATA(5).LT.1.) ) GO TO 9001
      IF(DATA(5).GT.1000.)  GO TO 9001
      IF(DATA(4).GT.4.)  GO TO 9001
      IF(DATA(4).LT.1.)  GO TO 9001
      IF( (DATA(3).LT.1.).OR.(DATA(3).GT.44.))  GO TO 9001
      IF( (DATA(4).EQ.4.).AND.(DATA(5).NE.0.) ) GO TO 9001
      IF( (DATA(7).LT.0.).OR.(DATA(8).LT.0.)) GO TO 9001
      IF( DATA(4).EQ.4. )  GO TO 8000
      INDEX=DATA(5)
      IF(  IDEV(INDEX).NE.0 ) GO TO 9005
      IDEV(INDEX)=1
      GO TO 8000
C
C GET HERE IF DCAB CARD READ
C
    4 DO 41 I=1,4
      ID=DATA(4)
      IF(ID.EQ.I) GO TO 42
   41 CONTINUE
      GO TO 9001
   42 IF(((DATA(4).EQ.3.).OR.(DATA(4).EQ.4.)).AND.(DATA(5).NE.0.)) GO TO
     1 9001
      IF( (DATA(3).LT.1.).OR.(DATA(3).GT.22.)) GO TO9001
      IF(((DATA(4).EQ.1.).OR.(DATA(4).EQ.2.)).AND.((DATA(5).LT.1.).OR.(D
     1ATA(5).GT.1000.)) ) GO TO 9001
      IF( (DATA(7).LT.0.).OR.(DATA(8).LT.0.))  GO TO 9001
      IF( ((DATA(4).EQ.2.).OR.(DATA(4).EQ.4.)).AND.(DATA(9).NE.0.) )
     1 GO TO 9001
      IF( ((DATA(4).EQ.1.).OR.(DATA(4).EQ.3.)).AND.(DATA(9).LE.0.) )
     1GO TO 9001
      IF( DATA(10).LT.0. ) GO TO 9001
      IF( (DATA(4).EQ.3.) .OR. (DATA(4).EQ.4.) )  GO TO 8000
      INDEX=DATA(5)
      IF( IDEV(INDEX).NE.0 ) GO TO 9005
      IDEV(INDEX)=1
```

```
      GO TO 8000
C
C GET HERE IF CAB CARD READ
C
    5 INDEX=DATA(3)
      IF( (DATA(3).LT.1.).OR.(DATA(3).GT.22.))  GO TO 9001
      IF( DATA(4).EQ.DATA(5) ) GO TO 9001
      IF( (DATA(4).GT.44.).OR.(DATA(5).GT.44.))  GO TO 9001
      IF( (DATA(4).LT.1.).OR.(DATA(5).LT.1.))  GO TO 9001
      IF( (DATA(7).LE.0.).OR.(DATA(8).LE.0.).OR.(DATA(9).LE.0.)
     1 .OR.(EX.LT.0.).OR.(DATA(10).LT.0.))  GO TO 9001
      IF((DATA(10).EQ.0.).AND.(EX.NE.0.))  GO TO 9001
      IF((DATA(10).NE.0.).AND.(EX.EQ.0.))  GO TO 9001
      IF( (NSEG.LT.1).OR.(NSEG.GT.50) ) GO TO 9001
      IF(ICAB(INDEX).NE.0) GO TO 9003
      ICAB(INDEX)=1
      INDEX=DATA(5)
      IF(ICHECK(INDEX).NE.0) GO TO 9002
      ICHECK(INDEX)=1
C
C GET HERE IF DATA OK
C
      INDEX=DATA(3)
      ZJUNC(INDEX)= DATA(4)
      LJUNC(INDEX)= DATA(5)
   51 NNODE(INDEX)= NSEG+1
      WCAB(INDEX)= DATA(6)
      CDCAB(INDEX)=DATA(7)
      DCAB(INDEX)=DATA(8)
      H(INDEX)= DATA(9)/NSEG
      ECICAB(INDEX)= DATA(10)
      EXPCAB(INDEX)= EX
      GO TO 8000
C
C GET HERE IF DEN CARD READ
C
    6 IF(DATA(3).LE.0.) GO TO 9001
      IFRHO=IFRHO+1
      IF( IFRHO.GT.1 )  GO TO 9006
      RHO= DATA(3)
      GO TO 1003
C
C GET HERE IF EOD CARD READ
C
    9 IFEOD=IFEOD+1
      DO 90 J=1,10
   90 DATAT(IROW,J)=DATA(J)
      IRMAX=IROW
      IF(FATE.NE.0.) RETURN
      IF(IFLG.EQ.1) GO TO 91
      GO TO 500
   91 IREAD=ISAV1
      IFLG=0
      GO TO 500
C
C GET HERE IF NJNC CARD READ
C
   10 IF( (DATA(3).LT.2.).OR.(DATA(3).GT.44.) )  GO TO 9001
      IFJNC=IFJNC+1
```

```
      IF(IFJNC.GT.1)  GO TO 9006
      NOJUNC=DATA(3)
      GO TO 1003
C
C GET HERE IF NDAT CARD READ
C
   12 NDATC=NDATC+1
      IF(NDATC.NE.IFLOD) GO TO 9018
      GO TO 998
12701 IFCOMP=0
      INDAT=0
      IFVEL=0
      IFANG=0
      IF((DATA(3).LT.0.).OR.(DATA(3).GT.2.))  GO TO 9001
      IVOPT=DATA(3)
      IF (NDATC.EQ.1) GO TO 1270
      IF (IVOPT.EQ.0) GO TO 1201
      IF (KFLG.EQ.0) GO TO 1271
      IF (IVOPT.EQ.KCUR) GO TO 1201
      GO TO 9117
 1270 IF (IVOPT.EQ.0) GO TO 1275
 1271 KFLG=1
      KCUR=IVOPT
      GO TO 1201
 1275 KFLG=0
C
C READ ONE INPUT RECORD FROM PARAMETRIC STUDY SOURCE DECK
C
 1201 READ(IREAD,1004) (DATN(I),I=1,10),EXX,NNSEG
      DO 1209 I=1,14
      IF(DATN(2).EQ.TEST(I)) GO TO (9018,1208,1208,1208,1208,9018,1207,
     1 1203,1260,9018,1202,9018,9018,9018),I
 1209 CONTINUE
      CALL SWTCH
      GO TO 9000
C
C GET HERE IF VEL CARD READ
C
 1202 IFVEL=IFVEL+1
      CALL SWTCH
      IF(IVOPT.EQ.0)  GO TO 9016
      IF(IFVEL.NE.1) GO TO 1225
      NVSEG=0
      DO 1224 I=1,25
      ZVEL(I)=0.
 1224 VELZ(I)=0.
 1225 NVSEG=NVSEG+1
      IF(NVSEG.GT.25)  GO TO 9006
      ZVEL(NVSEG)=DATA(3)
      DO 111 K=1,NVSEG
      IF((ZVEL(NVSEG).EQ.ZVEL(K)).AND.(K.NE.NVSEG))  GO TO 9006
  111 CONTINUE
      VELZ(NVSEG)=DATA(4)
      GO TO 1201
C
C GET HERE IF ANG CARD READ
C
 1203 IFANG=IFANG+1
      CALL SWTCH
```

```
      IF(IVOPT.EQ.0) GO TO 9018
      IF(IFANG.EQ.1)  NANG=0
      IF(DATA(4).LE.0.) GO TO 9001
      IF( DATA(5).LT.DATA(3) ) GO TO 9001
      THETAB=DATA(3)
      THETAS= DATA(4)
      THETAE= DATA(5)
      NANG=NANG+1
      IF( NANG.GT.1)  GO TO 9006
      GO TO 1201
C
C GET HERE IF COMP CARD READ
C
 1207 IFCOMP=IFCOMP+1
      CALL SWTCH
      IF(IFCOMP.EQ.1) NCOMP=0
      IF(DATA(3).LE.0.) GO TO 9001
       NCOMP= NCOMP+1
      IF( NCOMP.GT.1) GO TO 9006
      COMPD=DATA(3)
      GO TO 1201
C
C GET HERE IF PARAMETERS ARE BEING CHANGED
C LOCATE RECORD TO BE CHANGED AND BRANCH
C
 1208 IF(INDAT.NE.0) GO TO 12081
      READ(INTAPE,31)  ((DATAT(I,J),J=1,10),I=1,IRMAX)
      REWIND INTAPE
      INDAT=1
12081 DO 1290 I=1,IRMAX
      IROW=I
      IF((DATN(1).EQ.DATAT(IROW,1)).AND.(DATN(2).EQ.DATAT(IROW,2)))
     1 GO TO 1205
      IF(DATAT(IROW,2).EQ.TEST(9)) GO TO 9017
 1290 CONTINUE
 1205 DO 12051 J=1,10
12051 DATA(J)=DATAT(IROW,J)
      DO 1206 I=1,5
      IF(DATN(2).EQ.TEST(I)) GO TO (1206,1210,1220,1230,1240),I
 1206 CONTINUE
C
C GET HERE IF ANC CARD READ
C
 1210 IF(DATA(3).NE.DATN(3)) GO TO 9017
      CALL SWTCH
      INDEX=DATA(3)
      GO TO 22
C
C GET HERE IF DJNC CARD READ
C
 1220 DO 1222 I=3,5
      IF(DATA(I).NE.DATN(I)) GO TO 9017
 1222 CONTINUE
      CALL SWTCH
      IF((DATA(7).LT.0.).OR.(DATA(8).LT.0.)) GO TO 9001
      GO TO 8000
C
C GET HERE IF DCAB CARD READ
C
```

```
 1230 IF((DATA(3).NE.DATN(3)).OR.(DATA(5).NE.DATN(5))) GO TO 9017
      DO 1231 I=1,3,2
      F=I
      G=I+1
      IF(((DATA(4).EQ.F).OR.(DATA(4).EQ.G)).AND.((DATN(4).EQ.F).OR.(DATN
     1(4).EQ.G))) GO TO 1232
 1231 CONTINUE
      GO TO 9017
 1232 CALL SWTCH
      IF( ((DATA(4).EQ.2.).OR.(DATA(4).EQ.4.)).AND.(DATA(9).NE.0.) )
     1 GO TO 9001
      IF( (DATA(7).LT.0.).OR.(DATA(8).LT.0.))  GO TO 9001
      IF( ((DATA(4).EQ.1.).OR.(DATA(4).EQ.3.)).AND.(DATA(9).LE.0.) )
     1GO TO 9001
      IF( DATA(10).LT.0. ) GO TO 9001
      GO TO 8000
C
C GET HERE IF CAB CARD READ
C
 1240 DO 1241 I=3,5
      IF(DATA(I).NE.DATN(I)) GO TO 9017
 1241 CONTINUE
      CALL SWTCH
      EX=EXX
      NSEG=NNSEG
      INDEX=DATA(3)
      IF( (DATA(7).LE.0.).OR.(DATA(8).LE.0.).OR.(DATA(9).LE.0.)
     1 .OR.(EX.LT.0.).OR.(DATA(10).LT.0.))  GO TO 9001
      IF((DATA(10).EQ.0.).AND.(EX.NE.0.))  GO TO 9001
      IF((DATA(10).NE.0.).AND.(EX.EQ.0.))  GO TO 9001
      IF( (NSEG.LT.1).OR.(NSEG.GT.50) ) GO TO 9001
      GO TO 51
C
C GET HERE IF EOD CARD READ
C
 1260 IFEOD=IFEOD+1
      IF(FATE.NE.0.) RETURN
      GO TO 501
C
C GET HERE IF LUN CARD READ
C
   13 INTAPE=DATA(5)
      IFLG=DATA(6)
      IF((IFLG.LT.0).OR.(IFLG.GT.1)) GO TO 9001
      ISAV1=DATA(3)
      IF(IFLG.EQ.0) IREAD=DATA(3)
      IF(IFLG.EQ.1) IREAD=DATA(7)
      OFLG=DATA(8)
      IF((OFLG.LT.0).OR.(OFLG.GT.2)) GO TO 9001
      IF(OFLG.EQ.0) OUTAPE=DATA(4)
      IF(OFLG.NE.0) OUTAPE=DATA(9)
      IF((IPRNT.EQ.IREAD).OR.(IPRNT.EQ.INTAPE).OR.((OFLG.NE.0).AND.
     1(IPRNT.EQ.OUTAPE)).OR.(INTAPE.EQ.IREAD).OR.(INTAPE.EQ.OUTAPE).OR.
     2(IREAD.EQ.OUTAPE).OR.((IFLG.EQ.1).AND.(IREAD.EQ.ISAV1)))GO TO 9001
      GO TO 1003
   14 IF((IFEOD-NDATC).NE.1) GO TO 9018
      IF(OFLG.EQ.0) GO TO 141
      WRITE(OUTAPE,142) TEST(14)
  142 FORMAT(A4,I4,3E15.8)
```

```
  141 WRITE(IPRNT,140)
  140 FORMAT(1H1,18HANALYSIS COMPLETED)
      FATE=1.
      RETURN
C
C CHECK TO SEE IF SUSPENDED ARRAY SOURCE DECK COMPLETE
C
  500 IF((IFJNC.EQ.0).OR.(IFRHO.EQ.0).OR.(NANC.EQ.0)) GO TO 9007
  510 IF(FATE.EQ.0.)  GO TO 2000
      RETURN
C
C CHECK TO SEE IF PARAMETRIC STUDY SOURCE DECK COMPLETE
C
  501 IF((IVOPT.EQ.0).OR.(IVOPT.EQ.2)) GO TO 5101
      NZL=0
      IF(NVSEG.EQ.0) GO TO 9008
      IF(IFVEL.EQ.0) GO TO 549
C
C SORT VELOCITY PROFILE BY Z-COORDINATE
C
  550 DO 555 I=1,NVSEG
      K=I
      DO 555 J=K,NVSEG
      IF(ZVEL(I).LE.ZVEL(J)) GO TO 555
      TEMP=ZVEL(I)
      ZVEL(I)=ZVEL(J)
      ZVEL(J)=TEMP
      TEMP=VELZ(I)
      VELZ(I)=VELZ(J)
      VELZ(J)=TEMP
  555 CONTINUE
  549 INDEX=ANJUNC(1)
      ZANCM=PJUNC(3,INDEX)
      IF(NANC.EQ.1) GO TO 511
      DO 508 N=2,NANC
      INDEX=ANJUNC(N)
      IF(PJUNC(3,INDEX).LT.ZANCM) ZANCM=PJUNC(3,INDEX)
  508 CONTINUE
  511 DO 509 N=1,NVSEG
      IF(ZVEL(N).LE.ZANCM) NZL=NZL+1
  509 CONTINUE
      IF((NZL.EQ.0).OR.(NANG.EQ.0)) GO TO 9008
 5101 IF (NCOMP.EQ.0) GO TO 9008
      IF (INDAT.NE.0) GO TO 4000
  505 CONTINUE
      RETURN
C
C CHECK ON CONTINUITY OF CABLE NUMBERING AND COUNT CABLES
C
 2000 NCAB=ICAB(22)
      DO 2001 N=1,21
      NCAB=NCAB+ICAB(N)
      J=ICAB(N)-ICAB(N+1)
      IF (J.LT.0) GO TO 9009
 2001 CONTINUE
      IF (NCAB.EQ.0) GO TO 9009
C
C CHECK ON CONTINUITY OF JUNCTION NUMBERING AND COUNT JUNCTIONS
C
```

```
 2002 NJUNC=ICHECK(44)
      DO 2003 N=1,43
      NJUNC=NJUNC+ICHECK(N)
      J=ICHECK(N)-ICHECK(N+1)
      IF (J.LT.0) GO TO 9010
 2003 CONTINUE
C
C CHECK ON CONTINUITY OF DEVICE NUMBERING AND COUNT DEVICES
C
 2004 NDEV=IDEV(1000)
      DO 2005 N=1,999
      NDEV=NDEV+IDEV(N)
      J=IDEV(N)-IDEV(N+1)
      IF (J.LT.0) GO TO 9011
 2005 CONTINUE
 2007 IF (FATE.EQ.0.) GO TO 3000
      RETURN
C
C GET HERE IF ARRAY NUMBERED CORRECTLY
C CHECK TO SEE IF NIR CONSISTENT
C
 3000 NIRC=NCAB+NANC-NOJUNC
      IF (NIR.NE.NIRC) GO TO 9013
 3001 IF (FATE.EQ.0.) GO TO 4000
      RETURN
C
C GET HERE TO MAKE FINAL CHECK ON INTAPE
C
 4000 I=1
 4009 DO 4001J=1,10
 4001 DATA(J)=DATAT(I,J)
      I=I+1
      IF (DATA(2).EQ.TEST(1)) GO TO 4002
      IF (DATA(2).EQ.TEST(3)) GO TO 4003
      IF (DATA(2).EQ.TEST(4)) GO TO 4004
      IF (DATA(2).EQ. TEST(5)) GO TO 4005
      IF (DATA(2).EQ.TEST(9)) GO TO 4006
      GO TO 4009
C
C GET HERE FOR IR
C
 4002 ID1=DATA(3)
      ID2=DATA(4)
      IF((ID2.GT.NOJUNC).OR.(ID1.LE.NOJUNC).OR.(ID1.GT.NJUNC))GO TO 9014
      GO TO 4009
C
C GET HERE FOR DJNC
C
 4003 ID=DATA(3)
      IF(ID.GT.NJUNC) GO TO 9014
      GO TO 4009
C
C GET HERE FOR DCAB
C
 4004 ID=DATA(3)
      IF (ID.GT.NCAB) GO TO 9014
      RL=H(ID)*(NNODE(ID)-1)
      IF (DATA(10).GE.RL) GO TO 9014
      GO TO 4009
```

```
C
C GET HERE FOR CAB
C
 4005 ID=DATA(4)
      IF(ID.GT.NJUNC) GO TO 9014
      GO TO 4009
C
C GET HERE FOR EOD
C
 4006 WRITE(INTAPE,31) ((DATAT(I,J),J=1,10),I=1,IRMAX)
   31 FORMAT(F4.0,A4,8E15.8)
      REWIND INTAPE
      IF(NDATC.NE.0) RETURN
      IF (FATE.EQ.0.) GO TO 5000
      RETURN
C
C GET HERE IF ALL OK AND CALCULATE PATH
C K IS CABLE COUNTER
C
 5000 K=0
C
C LOOP=1 LOOKING FOR CABLES LEAVING ANCHORS
C
      LOOP=1
C
C JMINP REMEMBERS FIRST VALUE OF K ON A LEVEL OF TREE
C
      JMINP=1
C
C JMAX REMEMBERS NUMBER OF CABLES ON A LEVEL OF TOPOGRAPHIC TREE
C
      JMAX=NANC
      IF(LOOP.EQ.1) GO TO 5002
 5001 JMAX=K
      IF ((LOOP.EQ.1).AND.(K.NE.1)) GO TO 9015
      IF ((LOOP.EQ.2).AND.(JMIN.EQ.JMINP)) GO TO 9015
      IF (K.EQ.NCAB) GO TO 5008
      LOOP=2
 5002 JMIN=JMINP
      DO 5009 J=JMIN,JMAX
C
C LOOKING FOR CABLES LEAVING A JUNCTION
C
      DO 5007 N=1,NCAB
      GO TO (5003,5004),LOOP
 5003 IF(ZJUNC(N).EQ.ANJUNC(J)) GO TO 5005
      GO TO 5007
 5004 IPATHJ=PATH(J)
      IF(ZJUNC(N).EQ.LJUNC(IPATHJ)) GO TO 5005
      GO TO 5007
C
C GET HERE IF CABLE N STARTS AT JUNCTION M
C
 5005 K=K+1
      PATH(K)=N
      GO TO (5007,5006),LOOP
C
C REMEMBER HERE FIRST VALUE OF K ON TREE LEVEL
C
```

```
5006 IF(JMAX.EQ.(K-1)) JMINP=K
5007 CONTINUE
5009 CONTINUE
     GO TO 5001
5008 CONTINUE
     IF(FATE.NE.0.) RETURN
     WRITE(IPRNT,5010)
5010 FORMAT(///,5X,18HNO ERRORS DETECTED)
     GO TO 1003
C
C PUT DATA INTO DATA ARRAY
C
8000 IF(IFEOD.NE.0) GO TO 8003
     IF(((DATA(2).EQ.TEST(3)).OR.(DATA(2).EQ.TEST(4)))
    1.AND.(DATA(5).NE.0.)) GO TO 8001
     GO TO 8003
8001 ID=DATA(5)
     DO 8002 J=1,10
     DATAT(IROW,J)=DATAT(ID,J)
8002 DATAT(ID,J)=DATA(J)
     GO TO 8005
8003 DO 8004 J=1,10
8004 DATAT(IROW,J)=DATA(J)
     IF(IFEOD.NE.0) GO TO 1201
8005 IROW=IROW+1
     IF(IROW.LE.2150) GO TO 1003
     WRITE(IPRNT,8006)
8006 FORMAT(6X,2H17,18X,45HCOMMON/B1/ BOUND EXCEEDED.  SEE USERS MANUAL
    1.)
     GO TO 1003
C
C THIS SECTION GENERATES ALL ERROR MESSAGES
C
9116 FATE=1.
     WRITE(IPRNT,9500) IER,DATA(1),DATA(2)
9500 FORMAT(6X,I2,4X,F4.0,3X,A4)
     IF(IER.EQ.12) GO TO 4009
     IF(IER.EQ.16) RETURN
     IF(NDATC.EQ.0) GO TO 1003
     GO TO 1201
9000 IER=0
     GO TO 9116
9001 IER=1
     GO TO 9116
9002 IER=2
     GO TO 9116
9003 IER=3
     GO TO 9116
9004 IER=4
     GO TO 9116
9005 IER=5
     GO TO 9116
9006 IER=6
     GO TO 9116
9007 IER =7
     WRITE(IPRNT,9501) IER,IFJNC,IFRHO,NANC
9501 FORMAT(6X,I2,18X,5I3)
     FATE=1
     GO TO 510
```

```
9008 IER=14                                                             INP649
     FATE=1                                                             INP650
     WRITE(IPRNT,9501) IER,NCOMP,IVOPT,NVSEG,NZL,NANG                   INP651
     GO TO 505                                                          INP652
9009 IER=8                                                              INP653
     FATE=1                                                             INP654
     WRITE(IPRNT,9502) IER,(ICAB(I),I=1,22)                             INP655
     GO TO 2002                                                         INP656
9010 IER=9                                                              INP657
     FATE=1                                                             INP658
     WRITE(IPRNT,9502) IER,(ICHECK(I),I=1,44)                           INP659
9502 FORMAT(6X,I2,18X,44I2)                                             INP660
     GO TO 2004                                                         INP661
9011 IER=10                                                             INP662
     FATE=1                                                             INP663
     WRITE(IPRNT,9504) IER,(IDEV(I),I=1,1000)                           INP664
9504 FORMAT(6X,I2,12X,100I1,9(/,20X,100I1))                             INP665
     GO TO 2007                                                         INP666
9013 IER=11                                                             INP667
     FATE=1                                                             INP668
     WRITE(IPRNT,9501) IER,NCAB,NANC,NOJUNC,NIRC,NIR                    INP669
     GO TO 3001                                                         INP670
9014 IER=12                                                             INP671
     GO TO 9116                                                         INP672
9015 IER=13                                                             INP673
     FATE=1                                                             INP674
     WRITE(IPRNT,9507) IER                                              INP675
     GO TO 5008                                                         INP676
9507 FORMAT(6X,I2,18X,94HIMPROPER ARRAY REDUCTION OR JUNCTION NUMBERINGINP677
    1.  CHECK TREE REPRESENTATION OF ARRAY (SEE ARRAY,/,20X,83HREDUCTIOINP678
    2N SECTION OF USERS MANUAL) AGAINST JUNCTION NUMBERING ON ANC AND CINP679
    3AB CARDS.)                                                         INP680
9017 DATA(1)=DATN(1)                                                    INP681
     DATA(2)=DATN(2)                                                    INP682
9117 IER=15                                                             INP683
     GO TO 9116                                                         INP684
9018 IER=16                                                             INP685
     IF(IFEOD.EQ.0) GO TO 9116                                          INP686
     DATA(1)=DATN(1)                                                    INP687
     DATA(2)=DATN(2)                                                    INP688
     GO TO 9116                                                         INP689
     END                                                                INP690
```

```
      SUBROUTINE PHSOUT
C
C THIS ROUTINE GENERATES INFORMATION CONCERNING THE PHYSICAL
C CHARACTERISTICS OF THE STRUCTURAL CABLE ARRAY
C
      COMMON /B1/ FEJUNC,IR,DELTA1,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      WRITE(IPRNT,25)
   25 FORMAT(1H1, 54HPHYSICAL CHARACTERISTICS OF THE STRUCTURAL CABLE AR
     1RAY)
      WRITE(IPRNT,1) NANC
    1 FORMAT(///,19H NO. OF ANCHORS IS ,I2,//5X,64HJUNCTION NO.     X-COO
     1RDINATE        Y-COORDINATE        Z-COORDINATE)
      DO 2 N=1,NANC
      INDEX=ANJUNC(N)
    2 WRITE(IPRNT,3) INDEX,(PJUNC(I,INDEX),I=1,3)
    3 FORMAT(10X,I2,4X,3(5X,F10.2,3X))
      NOJUNC=NCAB+NANC-NIR
      WRITE(IPRNT,4) NOJUNC
    4 FORMAT(//,39H NO. OF JUNCTIONS IN ORIGINAL ARRAY IS ,I2)
      WRITE(IPRNT,5) NIR
    5 FORMAT(//,39H NO. OF CUTS MADE IN ORIGINAL ARRAY IS ,I2//1H ,2(5X,
     112HJUNCTION NO.),/,8X,6HOF CUT,6X,17HAT WHICH CUT MADE)
      IF(NIR.EQ.0) GO TO 30
      DO 6 N=1,NIR
    6 WRITE(IPRNT,7) IRJUNC(N),ERJUNC(N)
    7 FORMAT(10X,I2,15X,I2)
   30 WRITE(IPRNT,8) NCAB
    8 FORMAT(//,18H NO. OF CABLES IS ,I2,//5X,16HCABLE   S=0    S=L,43X,4H
     1DRAG,19X,22HCONSTITUTIVE      NO. OF,/,6X,3HNO.,1X,2(2X,4HJUNC),3X,
     26HLENGTH,03X,8HDIAMETER,3X,13HWEIGHT/LENGTH,3X,11HCOEFFICIENT,3X,
     38HRIGIDITY,6X,8HEXPONENT,5X,8HELEMENTS)
      DO 9 N=1,NCAB
      NSEG=NNODE(N)-1
      RL=H(N)*NSEG
    9 WRITE(IPRNT,10) N,ZJUNC(N),LJUNC(N),RL,DCAB(N),WCAB(N),CDCAB(N),
     1ECICAB(N),EXPCAB(N),NSEG
   10 FORMAT(6X,I2,5X,I2,4X,I2,1X,F9.1,3(3X,F7.3,4X),   F10.0,7X,F6.3,
     19X,I2 )
      WRITE(IPRNT,11)
   11 FORMAT(//,63H PROPERTIES OF THE DEVICES LOCATED AT JUNCTIONS ARE A
     1S FOLLOWS ,//6X,6HDEVICE,10X,6HDEVICE,9X,11HDEVICE DRAG,4X,14HDEVI
```

```
  2CE FRONTAL,/,5X,9HJUNC. NO.,8X,6HWEIGHT,9X,11HCOEFFICIENT,9X,      PHS060
  34HAREA)                                                             PHS061
 12 READ(INTAPE,13) (DATA(I),I=1,10)                                   PHS062
 13 FORMAT(F4.0,A4,8E15.8)                                             PHS063
    IF(DATA(2).EQ.TEST(3)) GO TO 14                                    PHS064
    IF(DATA(2).EQ.TEST(9)) GO TO 16                                    PHS065
    GO TO 12                                                           PHS066
 14 JUNC=DATA(3)                                                       PHS067
    WRITE(IPRNT,15) JUNC,(DATA(K),K=6,8)                               PHS068
 15 FORMAT(8X,I2,9X,F10.2,6X,F10.3,6X,F10.2)                           PHS069
    GO TO 12                                                           PHS070
 16 REWIND INTAPE                                                      PHS071
    WRITE(IPRNT,24) NDEV                                               PHS072
 24 FORMAT(//,32H TOTAL NO. OF INDEXED DEVICES IS,I4)                  PHS073
    WRITE(IPRNT,17) IVOPT                                              PHS074
 17 FORMAT(//,25H CURRENT FIELD OPTION IS ,I1)                         PHS075
    IV=IVOPT+1                                                         PHS076
    GO TO (22,18,22),IV                                                PHS077
 18 WRITE(IPRNT,19)                                                    PHS078
 19 FORMAT(/7X12HZ-COORDINATE,6X,11HVELOCITY OF,/,8X,10HOF CURRENT,7X,PHS079
   112HCURRENT AT Z)                                                   PHS080
    DO 20 N=1,NVSEG                                                    PHS081
 20 WRITE(IPRNT,21) ZVEL(N),VELZ(N)                                    PHS082
 21 FORMAT(2X,5X,F10.2,10X,F7.2)                                       PHS083
 22 WRITE(IPRNT,23) COMPD                                              PHS084
 23 FORMAT(//,38H ACCURACY REQUIRED IN CALCULATIONS IS ,F6.2)          PHS085
    IF(OFLG.NE.0) WRITE(IPRNT,111) NDATC                               PHS086
111 FORMAT(//,30H DEVICE LOCATION OUTPUT RECORD,I3,21H REFERS TO THIS PHS087
   1ARRAY  )                                                           PHS088
    RETURN                                                             PHS089
    END                                                                PHS090
```

```
      SUBROUTINE STROUT
C
C THIS ROUTINE GENERATES THE ARRAY STRUCTURAL OUTPUT
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      DIMENSION TEMP1(3),TEMP2(3),TEMP3(3),DISP(3),PSPACE(3)
      DIMENSION A(3),B(3),D(3),U(3),V(3),W(3),C(4),RQ(3),RI(3)
      NOJUNC=NCAB+NANC-NIR
      JUM=JUMP+1
      GO TO(100,200),JUM
  100 WRITE(IPRNT,1002)
 1002 FORMAT(1H1,33HARRAY EQUILIBRIUM WITH NO CURRENT ///)
      GO TO 300
  200 WRITE(IPRNT,1001) THETA
 1001 FORMAT(1H1,35HARRAY EQUILIBRIUM WITH CURRENT FROM,F8.2, 9H DEGREES
     1 ///)
C
C GENERATE ANCHOR HEADERS
C
  300 WRITE(IPRNT,1003)
 1003 FORMAT(1H ,13HARRAY ANCHORS    )
      WRITE(IPRNT,10031)
10031 FORMAT(1H ,13H----- -------/)
      WRITE(IPRNT,1004)
 1004 FORMAT(1H ,  32HJUNC. NO.  CABLE AT    TENSION AT    ,14X,26HFORCE C
     1OMPONENTS AT ANCHOR,13X,16HCABLE ANGLES WRT)
      WRITE(IPRNT,1005)
 1005 FORMAT(1H ,30HOF ANCHOR   ANCHOR       ANCHOR8X,6HX-COMP,6X,6HY-COM
     1P,6X,6HZ-COMP,3X,9HHOR.-COMP,5X,16HX-AXIS  XY-PLANE)
C
C THIS SECTION CALCULATES FORCES AND ANGLES AT ANCHORS
C
      DO 401 J1=1,NANC
      J3=0
  402 J2=1
  404 IF (ANJUNC(J1).EQ.ZJUNC(J2)) GO TO 403
  414 J2=J2+1
      IF(J2.LE.NCAB) GO TO 404
      IF(NIR.EQ.0) GO TO 401
  412 J3=1
  411 IF(ANJUNC(J1).EQ.ERJUNC(J3)) GO TO 405
```

```
  413 J3=J3+1
      IF(J3.LE.NIR)   GO TO 411
      GO TO 401
  405 INDEX=IRJUNC(J3)
      DO 406 J4=1,NCAB
      IF(INDEX.EQ.LJUNC(J4)) GO TO 407
  406 CONTINUE
      GO TO 401
  407 MM=J4
      MX=NNODE(J4)
      T=TCAB(MX,MM)
      RX=-RCAB(1,MX,MM)
      RY=-RCAB(2,MX,MM)
      RZ=-RCAB(3,MX,MM)
      GO TO 408
  403 MM=J2
      T=TCAB(1,MM)
      RX=RCAB(1,1,MM)
      RY=RCAB(2,1,MM)
      RZ=RCAB(3,1,MM)                            RH2 = RH
  408 RH=SQRT(RX**2 + RY**2)                     IF(RH2.EQ.0) RH=1.
      A1=ASIN(RY/RH)/PIP
      A2=ASIN(RZ/T)/PIP                          IF(RH2.EQ.0) RH=0.
      IF((RX.LT.0.).AND.(RY.GE.0.)) A1=180.-A1
      IF((RX.LT.0.).AND.(RY.LT.0.)) A1=-180.-A1
      WRITE(IPRNT,1006) ANJUNC(J1),MM,T,RX,RY,RZ,RH,A1,A2
 1006 FORMAT(1H ,3X,I2,9X,I2,5X,F10.1,3X,4(F10.1,2X),2(2X,F7.2,1X))
      IF(J2.LT.NCAB) GO TO 414
      IF( (J2.EQ.NCAB).AND.(J3.EQ.0) ) GO TO 412
      IF( (J2.EQ.NCAB).AND.(J3.NE.NJUNC) ) GO TO 413
  401 CONTINUE
C
C GENERATE CABLE HEADERS
C
      WRITE(IPRNT,2000)
 2000 FORMAT(1H ,///1H ,12HARRAY CABLES )
      WRITE(IPRNT,2001)
 2001 FORMAT(1H ,12H----- ------/)
      WRITE(IPRNT,2002)
 2002 FORMAT(1H ,120HCABLE  MAXIMUM  S-COORD  MINIMUM  S-COORD  MAXIMUM
     1 S-COORD     LOCATION OF THIS POINT      NO CURRENT LOC. OF THIS PO
     2INT )
      WRITE(IPRNT,2003)
 2003 FORMAT(1H ,120H NO.   TENSION    OF     TENSION     OF       DISP.
     1   OF        X-COORD    Y-COORD    Z-COORD    X-COORD    Y-COORD    Z-COO
     2RD   )
C
C THIS SECTION CALCULATES MAXIMUM AND MINIMUM CABLE TENSIONS
C AND MAXIMUM CABLE DISPLACEMENTS FROM NO CURRENT LOCATION
C BY EXTRAPOLATION BETWEEN CABLE NODES
C
C INITIALIZE EXTREMA
C
      DO 799 NN=1,NCAB
      N=NN
      TMAX=TCAB(1,N)
      STMAX=0.
      TMIN=TMAX
      STMIN=0.
```

```
      GO TO (802,801),JUM
  801 DMAX=0.
      DO 700 I=1,3
  700 DMAX=DMAX+(PCAB(I,1,N)-PCABO(I,1,N))**2
      DMAX=SQRT(DMAX)
      SDMAX=0.
  802 MX=NNODE(N)-1
      DO 750 MM=1,MX
      M=MM
      M1=MM+1
C
C CALCULATE EXTRAPOLATION QUANTITIES
C
      RR=0.
      RD=0.
      DD=0.
      DO 701 I=1,3
      D(I)=(RCAB(I,M1,N)-RCAB(I,M,N))/H(N)
      GO TO  (804,803) ,JUM
  803 A(I)=EXCAB(M,N)*RCAB(I,M,N)/TCAB(M,N)
      B(I)=(EXCAB(M1,N)*RCAB(I,M1,N)/TCAB(M1,N)-A(I))/H(N)
      U(I)=PCAB(I,M,N)-PCABO(I,M,N)
  804 RR=RR+RCAB(I,M,N)*RCAB(I,M,N)
      RD=RD+RCAB(I,M,N)*D(I)
  701 DD=DD+D(I)*D(I)
      GO TO (806,805),JUM
  805 DO 702 I=1,3
      TEMP1(I)=RCAB(I,M,N)
      TEMP2(I)=RCAB(I,M1,N)
      RCAB(I,M,N)=RCABO(I,M,N)
  702 RCAB(I,M1,N)=RCABO(I,M1,N)
      UU=0.
      UV=0.
      UVW=0.
      VW=0.
      WW=0.
      DO 703 I=1,3
      V(I)=EXCAB(M,N)*RCAB(I,M,N)/TCAB(M,N)
      W(I)=(EXCAB(M1,N)*RCAB(I,M1,N)/TCAB(M1,N)-V(I))/H(N)
      V(I)=A(I)-V(I)
      W(I)=(B(I)-W(I))/2.
      UU= UU+U(I)*U(I)
      UV=UV+U(I)*V(I)
      UVW=UVW+2.*U(I)*W(I)+V(I)*V(I)
      VW=VW+V(I)*W(I)
  703 WW=WW+W(I)*W(I)
      DO 704 I=1,3
      RCAB(I,M,N)=TEMP1(I)
  704 RCAB(I,M1,N)=TEMP2(I)
  806 C(4)=0.
      C(3)=0.
      C(2)=DD
      C(1)=RD
      CALL RPOLY(C,RO,RI)
C
C CALCULATE TENSION EXTREMA IN SEGMENT
C
      JTIME=0
      DO 719 I=1,3
```

```
      IF( RI(I).NE.0.) GO TO 719
      IF((RQ(I).LE.0.).OR.(RQ(I).GE.H(N))) GO TO 719
      SIG=RQ(I)
  710 TE=SQRT(RR+2.*RD*SIG+DD*SIG**2)
      IF(TE.GT.TMAX)     GO TO 712
      IF(TE.LT.TMIN) GO TO 713
  711 IF(JTIME.EQ.0) GO TO 719
      GO TO 715
  712 TMAX=TE
      STMAX=H(N)*(M-1)+SIG
      GO TO 711
  713 TMIN=TE
      STMIN=H(N)*(M-1)+SIG
      GO TO 711
  719 CONTINUE
      JTIME=1
      SIG=H(N)
      GO TO 710
  715 GO TO(750,716),JUM
  716 C(4)=2.*WW
      C(3)=3.*VW
      C(2)=UVW
      C(1)=1
      CALL RPOLY(C,RQ,RI)
C
C CALCULATE MAXIMUM DISPLACEMENT IN SEGMENT
C
      JTIME=0
      DO 729 I=1,3
      IF (RI(I).NE.0.) GO TO 729
      IF((RQ(I).LE.0.).OR.(RQ(I).GE.H(N))) GO TO 729
      SIG=RQ(I)
  720 DE=SQRT(UU+2.*UV*SIG+UVW*SIG**2+2.*VW*SIG**3+WW*SIG**4)
      IF(DE.GT.DMAX) GO TO 722
  721 IF(JTIME.EQ.0) GO TO 729
      GO TO 750
  722 DMAX=DE
      SDMAX= H(N)*(M-1)+SIG
      GO TO 721
  729 CONTINUE
      JTIME=1
      SIG=H(N)
      GO TO 720
  750 CONTINUE
C
C EXTREMA ALONG A CABLE NOW DETERMINED
C CALCULATE FINAL AND INITIAL COORDINATES OF MAXIMALLY DISPLACED POINT
C
      GO TO (810,809),JUM
  809 RL=H(N)*MX
      IF(SDMAX.LT.RL) GO TO 760
      K=LJUNC(N)
      DO 751 I=1,3
      A(I)=PJUNC(I,K)
  751 B(I)=PJUNCO(I,K)
      GO TO 780
  760 DATA(2)=TEST(4)
      DATA(3)=N
      DATA(10)=SDMAX
```

```
      J=(SDMAX/H(N))+1
      J1=J+1
      DO 761 II=1,3
      I=II
  761 A(I)=SPACE(I)
      DO 762 I=1,3
      TEMP1(I)=RCAB(I,J,N)
      TEMP2(I)=RCAB(I,J1,N)
      TEMP3(I)=PCAB(I,J,N)
      RCAB(I,J,N)=RCABO(I,J,N)
      RCAB(I,J1,N)=RCABO(I,J1,N)
  762 PCAB(I,J,N)=PCABO(I,J,N)
      DO 763 II=1,3
      I=II
  763 B(I)=SPACE(I)
      DO 764 I=1,3
      RCAB(I,J,N)=TEMP1(I)
      RCAB(I,J1,N)=TEMP2(I)
  764 PCAB(I,J,N)=TEMP3(I)
  780 CONTINUE
      WRITE(IPRNT,781)N,TMAX,STMAX,TMIN,STMIN,DMAX,SDMAX,(A(I),I=1,3),
     1 (B(I),I=1,3)
      GO TO 799
  810 WRITE(IPRNT,781)N,TMAX,STMAX,TMIN,STMIN
  781 FORMAT(1H ,I4,1X,6F9.1,6F10.1)
  799 CONTINUE
C
C GENERATE JUNCTION HEADERS
C
      WRITE(IPRNT,1007)
 1007 FORMAT(1H ///1H ,15HARRAY JUNCTIONS)
      WRITE(IPRNT,10071)
10071 FORMAT(1H ,15H----- --------- / )
      WRITE(IPRNT,1008)
 1008 FORMAT(1H ,27HJUNC.  CABLE AT  TENSION AT,3X,16HCABLE ANGLES WRT,
     110X,17HJUNCTION LOCATION,10X,33HDISPLACEMENT FROM NO CURRENT LOC.)
      WRITE(IPRNT,1009)
 1009 FORMAT(1H ,15H NO.   JUNCTION3X,8HJUNCTION4X,16HX-AXIS  XY-PLANE,
     168H    X-COORD    Y-COORD    Z-COORD       X-DISP     Y-DISP     Z-
     2DISP  )
C
C THIS SECTION CALCULATES JUNCTION FORCES, LOCATIONS AND DISPLACEMENTS
C
      IF=0
      DO 601 J1=1,NOJUNC
      DO 605 K=1,NANC
      IF(ANJUNC(K).EQ.J1) GO TO 601
  605 CONTINUE
      GO TO (602,603),JUM
  603 DO 604 J2=1,3
  604 DISP(J2)=PJUNC(J2,J1)-PJUNCO(J2,J1)
  602 J6=1
  608 IF(J1.EQ.ZJUNC(J6)) GO TO 606
  615 IF(J1.EQ.LJUNC(J6)) GO TO 607
  617 J6=J6+1
      IF(J6.LE.NCAB) GO TO 608
      IF(NIR.EQ.0) GO TO 601
      J7=1
  609 IF(J1.EQ.ERJUNC(J7)) GO TO 610
```

```
  616 J7=J7+1                                                          STR296
      IF(J7.LE.NIR)    GO TO 609                                       STR297
      GO TO 601                                                        STR298
  606 MM=J6                                                            STR299
      T=TCAB(1,MM)                                                     STR300
      RX=RCAB(1,1,MM)                                                  STR301
      RY=RCAB(2,1,MM)                                                  STR302
      RZ=RCAB(3,1,MM)                                                  STR303
      GO TO 611                                                        STR304
  610 INDEX=IRJUNC(J7)                                                 STR305
      DO 612 J8=1,NCAB                                                 STR306
      IF(INDEX.EQ.LJUNC(J8))GO TO 613                                  STR307
  612 CONTINUE                                                         STR308
  613 MM=J8                                                            STR309
      GO TO 614                                                        STR310
  607 MM=J6                                                            STR311
  614 MX=NNODE(MM)                                                     STR312
      T=TCAB(MX,MM)                                                    STR313
      RX=-RCAB(1,MX,MM)                                                STR314
      RY=-RCAB(2,MX,MM)                                                STR315
      RZ=-RCAB(3,MX,MM)                                                STR316
  611 RH=SQRT(RX**2+RY**2)       IF(RH.EQ.0) RH=1.                     STR317
      A1=ASIN(RY/RH)/PIP                                               STR318
      A2=ASIN(RZ/T)/PIP                                                STR319
      IF((RX.LT.0.).AND.(RY.GE.0.)) A1=180.-A1                         STR320
      IF((RX.LT.0.).AND.(RY.LT.0.)) A1=-180.-A1                        STR321
      IF((IF.EQ.0).AND.(JUM.EQ.1)) WRITE(IPRNT,620) J1,MM,T,A1,A2,(PJUNCSTR322
     1(K,J1),K=1,3)                                                    STR323
      IF((IF.EQ.0).AND.(JUM.EQ.2)) WRITE(IPRNT,620) J1,MM,T,A1,A2,(PJUNCSTR324
     1(K,J1),K=1,3) ,(DISP(I),I=1,3)                                   STR325
      IF(IF.EQ.1) WRITE(IPRNT,620) J1,MM,T,A1,A2                       STR326
      IF=1                                                             STR327
  620 FORMAT(1H ,I4,6X,I2,4X,F10.1,3X,F7.2,3X,F7.2,3(1X,F10.1),1X,3(1X, STR328
     1F10.1))                                                          STR329
      IF((J6.LT.NCAB).AND.(J1.EQ.ZJUNC(J6))) GO TO 615                 STR330
      IF((J6.LT.NCAB).AND.(J1.EQ.LJUNC(J6))) GO TO 617                 STR331
  601 IF=0                                                             STR332
C                                                                      STR333
C GENERATE INDEXED DEVICE HEADERS                                      STR334
C                                                                      STR335
      WRITE(IPRNT,519 )                                                STR336
  519 FORMAT(1H ,///1H ,34HINDEXED DEVICES ALONG ARRAY CABLES  )       STR337
      WRITE(IPRNT,5191)                                                STR338
 5191 FORMAT(1H ,34H------- ------- -  -- --- - *      /)              STR339
      WRITE(IPRNT,520 )                                                STR340
  520 FORMAT(1H ,7HDEVICE  ,5HCABLE,6X,1HS,7X,10HTENSION AT,13X,15HDEVICESTR341
     1 LOCATION,12X,36H   DISPLACEMENT FROM NO CURRENT LOC. )          STR342
      WRITE(IPRNT,521 )                                                STR343
  521 FORMAT(1H ,109HINDEX    NO.    COORDINATE    DEVICE        X-COORD STR344
     1  Y-COORD       Z-COORD            X-DISP      Y-DISP      Z-DISP) STR345
C                                                                      STR346
C THIS SECTION CALCULATES TENSIONS AT INDEXED DEVICES,                 STR347
C DEVICE LOCATIONS, AND DEVICE DISPLACEMENTS                           STR348
C                                                                      STR349
      IF (NDEV.EQ.0) GO TO 511                                         STR350
      DO 510 N=1,NDEV                                                  STR351
      READ (INTAPE,501) (DATA(I),I=1,10)                               STR352
  501 FORMAT (I4,A4,8E15.8)                                            STR353
      IF (DATA(2).EQ.TEST(3)) GO TO 510                                STR354
```

```
      K=DATA(3)
      M=DATA(10)/H(K) + 1
      INDEX=DATA(5)
      SIGMA=DATA(10)-(M-1)*H(K)
      TEN2=0.
      DO 502 II=1,3
      I=II
      PSPACE(I)=SPACE(I)
  502 TEN2=TEN2+(RCAB(I,M,K)+(RCAB(I,M+1,K)-RCAB(I,M,K))*SIGMA/H(K))**2
      TEN=SQRT(TEN2)
      GO TO (503,505),JUM
  503 WRITE (IPRNT,509) INDEX,K,DATA(10),TEN,(PSPACE(I),I=1,3)
      GO TO 510
  505 DO 506 I=1,3
      TEMP1(I)=RCAB(I,M,K)
      TEMP2(I)=RCAB(I,M+1,K)
      TEMP3(I)=PCAB(I,M,K)
      RCAB(I,M,K)=RCABO(I,M,K)
      RCAB(I,M+1,K)=RCABO(I,M+1,K)
  506 PCAB(I,M,K)=PCABO(I,M,K)
      DO 507 II=1,3
      I=II
  507 DISP(I)=PSPACE(I)-SPACE(I)
      DO 508 I=1,3
      RCAB(I,M,K)=TEMP1(I)
      RCAB(I,M+1,K)=TEMP2(I)
  508 PCAB(I,M,K)=TEMP3(I)
      WRITE(IPRNT,509) INDEX,K,DATA(10),TEN,(PSPACE(I),I=1,3),
     1(DISP(I),I=1,3)
  509 FORMAT(1H ,I4,4X,I2,4X,F9.1,3X,F9.1,1X,3(2X,F10.1),4X,3(1X,F10.1))
  510 CONTINUE
  511 REWIND INTAPE
      RETURN
      END
```

```
      SUBROUTINE TAPOUT
C
C THIS ROUTINE GENERATES THE TAPE OR CARDS GIVING
C THE LOCATIONS OF THE INDEXED DEVICES
C
      COMMON /B1/ FEJUNC,IR,DELTA1,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NI-,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      DIMENSION PSPACE(3)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      ID1=4H CUR
      ID2=4H DEV
      ID3=4H REC
    1 FORMAT(A4,I4,3F10.2)
      JUM=JUMP+1
      GO TO (2,3) JUM
    2 WRITE(OUTAPE,1) ID3,NDATC
      WRITE(OUTAPE,1) ID1,JUMP
      GO TO 4
    3 WRITE(OUTAPE,1)ID1,JUMP,THETA
    4 IF (NDEV.EQ.0) GO TO 8
      DO 7 NN=1,NDEV
      READ(INTAPE,5) (DATA(K),K=1,10)
    5 FORMAT(F4.0,A4,8E15.8)
      INDEX=DATA(5)
C
C CALCULATE LOCATION OF DEVICE IN SPACE
C
      DO 6 J=1,3
      I=J
    6 PSPACE(I)=SPACE(I)
      WRITE(OUTAPE,1)ID2,INDEX,(PSPACE(I),I=1,3)
    7 CONTINUE
    8 CONTINUE
      REWIND +NTAPE
      RETURN
      END
```

```
      SUBROUTINE ERROR
C
C THIS ROUTINE GENERATES A TYPE 18 ERROR MESSAGE
C GET HERE IF ALL IMAGINARY REACTIONS DO NOT CHANGE
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAL,COMPD,THETAG,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      E=E*10.
      WRITE(IPRNT,1)
    1 FORMAT(1H1,52H  TYPE 18 ERROR.  STRUCTURAL ANALYSIS NOT COMPLETED.
     1/60H  PRINTOUT GIVEN FOR DIAGNOSTIC PURPOSES.  SEE USERS MANUAL./)
      JUM=JUMP+1
      GO TO (4,6),JUM
    4 WRITE(IPRNT,5)
    5 FORMAT(5X,40HEXISTING CURRENT CONDITION IS NO CURRENT  /)
      GO TO 8
    6 WRITE(IPRNT,7) THETA
    7 FORMAT(5X,38HEXISTING CURRENT CONDITION IS THETA = ,F3.0/)
    8 WRITE(IPRNT,9) E
    9 FORMAT(5X,35HBEST VALUE OF ACCURACY OBTAINED IS ,F6.2)
      DO 2  N=1,NCAB
      NN=N
      WRITE(IPRNT,10) NN
   10 FORMAT(//5X,29HTHE TENSIONS IN CABLE NUMBER ,I2,4H ARE,/)
      L=NNODE(N)
      DO 2  M=1,L
      MM=M
      T=TCAB(MM,NN)
      S=H(NN)*(MM-1.)
    2 WRITE(IPRNT,3) T,S
    3 FORMAT(7X,2HT=,F10.2,6H AT S=,F10.2)
      RETURN
      END
```

```
      SUBROUTINE RPOLY(C,RR,RI)
C
C THIS ROUTINE FINDS REAL ROOTS OF POLYNOMIAL EQUATIONS UP TO
C C(4)*(X**3) + C(3)*(X**2) + C(2)*X + C(1) = 0  FOR USE IN
C EVALUATING MAXIMUM CABLE DISPLACEMENTS AND TENSION EXTREMA
C
C REAL PARTS OF ROOTS ARE PLACED IN RR, IMAGINARY PARTS IN RI
C SINCE ONLY REAL ROOTS ARE OF INTEREST, ALL NON-REAL OR
C NON-EXISTING ROOTS RETURN RR=0 , RI=1
C
      DIMENSION C(4),RR(3),RI(3)
      IF(C(4).NE.0.) GO TO 30
      IF(C(3).NE.0.) GO TO 20
      IF(C(2).NE.0.) GO TO 10
C
C GET HERE IF EQUATION IDENTICALLY SATISFIED
C
  100 DO 1 I=1,3
      RR(I)=0.
    1 RI(I)=1.
      RETURN
C
C GET HERE IF EQUATION LINEAR
C
   10 RR(1)=-C(1)/C(2)
      RI(1)=0.
  200 DO 11 I=2,3
      RR(I)=0.
   11 RI(I)=1.
      RETURN
C
C GET HERE IF EQUATION QUADRATIC
C
   20 DISC=C(2)**2-4.*C(1)*C(3)
      IF(DISC.GE.0.) GO TO 21
      GO TO 100
   21 RR(1)=(-C(2)+SQRT(DISC))/(2.*C(3))
      RI(1)=0.
      RR(2)=(-C(2)-SQRT(DISC))/(2.*C(3))
      RI(2)=0.
      RR(3)=0.
      RI(3)=1.
      RETURN
C
C GET HERE IF EQUATION CUBIC
C
   30 P=C(3)/C(4)
      Q=C(2)/C(4)
      R=C(1)/C(4)
      A=(3.*Q-P**2)/3.
      B=(2.*P**3-9.*P*Q+27.*R)/27.
      DISC= (B**2)/4.+(A**3)/27.
      IF(DISC)32,31,31
   31 DISC= SQRT(DISC)
      CAPA=(((-B/2.+DISC)**4)**(1./3.))/(-B/2.+DISC)
      CAPB=(((-B/2.-DISC)**4)**(1./3.))/(-B/2.-DISC)
      RR(1)=CAPA+CAPB-P/3.
      RI(1)=0.
      IF(DISC.GT.0.)   GO TO 200
```

```
      RR(2)=-CAPA-P/3.                                                  RP0060
      RI(2)=0.                                                          RP0061
      RR(3)=RR(2)                                                       RP0062
      RI(3)=0.                                                          RP0063
      RETURN                                                            RP0064
   32 DISC=2.*SQRT(-A/3.)                                               RP0065
      PHI=ACOS((+3.*B)/(A*DISC))/3.                                     RP0066
      RR(1)=-P/3.+DISC*COS(PHI)                                         RP0067
      RR(2)=-P/3.+DISC*COS(PHI+3.14159265*(2./3.))                      RP0068
      RR(3)=-P/3.+DISC*COS(PHI+3.14159265*(4./3.))                      RP0069
      RI(1)=0.                                                          RP0070
      RI(2)=0.                                                          RP0071
      RI(3)=0.                                                          RP0072
      RETURN                                                            RP0073
      END                                                               RP0074
```

```
      SUBROUTINE SWTCH                                                  SWT001
C                                                                       SWT002
C THIS ROUTINE SWITCHES INPUT DATA                                      SWT003
C                                                                       SWT004
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,SWT005
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO                        SWT006
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,           SWT007
     1CDCAB,DCAB,FATE,NANG,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG, SWT008
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,               SWT009
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,    SWT010
     4WCAB,IDEV,ICHECK,NDEV,NDATC                                       SWT011
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44SWT012
     1)                                                                 SWT013
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)  SWT014
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)            SWT015
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22) SWT016
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)      SWT017
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)       SWT018
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)         SWT019
      DIMENSION ICHECK(44)                                              SWT020
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG                          SWT021
      INTEGER PATH                                                      SWT022
      REAL IR,IRS                                                       SWT023
      DO 1 I=1,10                                                       SWT024
    1 DATA(I)=DATN(I)                                                   SWT025
      RETURN                                                            SWT026
      END                                                               SWT027
```

```
      SUBROUTINE START
C
C THIS ROUTINE CALCULATES THE INITIAL GUESSES AT THE IMAGINARY
C AND EQUILIBRATING REACTIONS AND THE INITIAL DELTA BASED ON
C THE TOTAL WEIGHT OF THE ARRAY
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      WEIGHT=0.
      DO 1 J=1,NJUNC
    1 WEIGHT=WEIGHT+FEJUNC(3,J)
      DO 2 N=1,NCAB
      INNN=NNODE(N)-1
      DO 2 M=1,INNN
    2 WEIGHT=WEIGHT+ FCAB(3,M,N)
      DO 3 N=1,NIR
      KER=ERJUNC(N)
      DO 3 I=1,3
    3 IR(I,KER)=0.
      DO 4 N=1,NIR
      KIR=IRJUNC(N)
      KER=ERJUNC(N)
      IR(1,KIR)=0.
      IR(2,KIR)=0.
      IR(3,KIR)=-WEIGHT/(NIR+1.101)
    4 IR(3,KER)=IR(3,KER)-IR(3,KIR)
      DELTAI= ABS(WEIGHT)/(NIR+1)
      RETURN
      END
```

```
      FUNCTION TCAB(M,K)
C
C THIS ROUTINE CALCULATES THE TENSION AT NODE M OF CABLE K
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANG,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      TCAB     =SQRT(RCAB(1,M,K)**2 + RCAB(2,M,K)**2 + RCAB(3,M,K)**2 )
      RETURN
      END
```

```
      FUNCTION EXCAB(M,K)
C
C THIS ROUTINE CALCULATES (1 + STRAIN) AT NODE M OF CABLE K
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANG,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      IF(EXPCAB(K).EQ.0.) GO TO 1
      EXCAB=1.+(TCAB(M,K)/ECICAB(K))**EXPCAB(K)
      RETURN
    1 EXCAB=1.
      RETURN
      END
```

```
      FUNCTION EFORCE(I)
C
C THIS ROUTINE CALCULATES THE DEVICE FORCES IN DIRECTION I
C USING THE NORMAL DRAG APPROXIMATION FOR IN-LINE DEVICES
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      DIMENSION WTEL(3),VNORM(3),PSPACE(3)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
C
C CALCULATE THE WEIGHT VECTOR OF A DEVICE
C
      WTEL(1)=0.
      WTEL(2)=0.
      WTEL(3)=DATA(6)
C
C CHECK TO SEE IF CURRENT OR NO CURRENT
C
      JUM=JUMP+1
      GO TO(1,2),JUM
C
C GET HERE IF NO CURRENT
C
    1 EFORCE = WTEL(I)
      RETURN
C
C GET HERE IF CURRENT
C CALCULATE LOCATION OF DEVICE IN SPACE
C
    2 DO 3 KK=1,3
      K=KK
    3 PSPACE(K)=SPACE(K)
C
C CHECK IF DEVICE IS IN-LINE OR FREE
C
      IJMP=DATA(4)
      GO TO (5,4,5,4),IJMP
C
C GET HERE IF FREE TYPE DEVICE -- CALCULATE MAGNITUDE OF THE CURRENT
C
    4 VMAG=SQRT(VELOC(1,PSPACE)**2 + VELOC(2,PSPACE)**2 + VELOC(3,PSPACE
     1)**2 )
C
```

```
C CALCULATE THE FORCE ON THE FREE DEVICE
C
      EFORCE = WTEL(I)+(RHO/2.)*DATA(7)*DATA(8)*VMAG*VELOC(I,PSPACE)
      RETURN
C
C GET HERE IF IN-LINE DEVICE
C CALCULATE THE TANGENTIAL PROJECTION OF THE CURRENT ON THE DEVICE
C TANG(I) EVALUATES THE UNIT TANGENT TO A CABLE AT ANY POINT
C
    5 VPROJ=0.
      DO 6 KK=1,3
      K=KK
    6 VPROJ=VPROJ + VELOC(K,PSPACE)*TANG(K)
C
C CALCULATE THE NORMAL COMPONENT OF THE CURRENT AND ITS MAGNITUDE
C
      DO 7 KK=1,3
      K=KK
    7 VNORM(K)=VELOC(K,PSPACE)-VPROJ*TANG(K)
      VNMAG=SQRT(VNORM(1)**2 + VNORM(2)**2 + VNORM(3)**2)
C
C CALCULATE THE FORCE ON THE IN-LINE DEVICE
C
      EFORCE=WTEL(I)+(RHO/2.)*DATA(7)*(DATA(8)/12.)*DATA(9)*VNMAG
     1 *VNORM(I)
      RETURN
      END
```

```
      FUNCTION CFORCE (I,M,N)
C
C THIS ROUTINE CALCULATES THE FORCE/LENGTH IN DIRECTION I AT NODE M
C ON CABLE N USING THE NORMAL DRAG FORCE APPROXIMATION
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      DIMENSION WTCAB(3),VNORM(3),PSPACE(3)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
C
C CALCULATE THE WEIGHT/LENGTH VECTOR
C
      WTCAB(1)=0.
      WTCAB(2)=0.
      WTCAB(3)=WCAB(N)
C
C CHECK TO SEE IF CURRENT OR NO CURRENT
C
      JUM=JUMP+1
      GO TO (1,2),JUM
C
C GET HERE IF NO CURRENT
C
    1 CFORCE=WTCAB(I)
      RETURN
C
C GET HERE IF CURRENT
C CALCULATE LOCATION OF NODE IN SPACE
C
    2 DO 3 K=1,3
    3 PSPACE(K)=PCAB(K,M,N)
C
C CALCULATE THE TANGENTIAL PROJECTION OF THE CURRENT ON THE CABLE
C
      VPROJ=0.
      DO 4 KK=1,3
      K=KK
    4 VPROJ=VPROJ + VELOC(K,PSPACE) * RCAB(K,M,N) / TCAB(M,N)
C
C CALCULATE THE NORMAL COMPONENT OF THE CURRENT AND ITS MAGNITUDE
C
      DO 5 KK=1,3
      K=KK
```

```
  5 VNORM(K)= VELOC(K,PSPACE) - VPROJ * RCAB(K,M,N) / TCAB(M,N)
    VNMAG = SQRT(VNORM(1)**2 + VNORM(2)**2 + VNORM(3)**2 )
C
C CALCULATE THE FORCE/LENGTH
C
    CFORCE=WTCAB(I)+(RHO/2.)*CDCAB(N)*(DCAB(N)/12.)*EXCAB(M,N)*VNMAG
   1 *VNORM(I)
    RETURN
    END
```

```
      FUNCTION SPACE(I)
C
C THIS ROUTINE CALCULATES THE LOCATION IN SPACE
C OF ANY POINT ON THE ARRAY
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAD,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDLV,ICHECK,NDLV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      IF(DATA(2).EQ.TEST(3))  GO TO 1
      IF(DATA(2).EQ.TEST(4))  GO TO 2
C
C GET HERE IF JUNCTION POINT
C
    1 K=DATA(3)
      SPACE = PJUNC(I,K)
      RETURN
C
C GET HERE IF POINT ON A CABLE
C
    2 N=DATA(3)
      M=    (DATA(10)/H(N)) +1
C
C CALCULATE DISTANCE, SIGMA, OF POINT FROM NODE M
C
      SIGMA = DATA(10) - (M-1)*H(N)
C
C CALCULATE EXTRAPOLATION QUANTITIES
C
      EM=EXCAB(M,N)*RCAB(I,M,N)/TCAB(M,N)
      EM1=EXCAB(M+1,N)*RCAB(I,M+1,N)/TCAB(M+1,N)
C
C CALCULATE LOCATION
C
      SPACE = PCAB(I,M,N)+EM*SIGMA+((EM1-EM)/H(N))*(SIGMA**2)/2.
      RETURN
      END
```

```
      FUNCTION VELOC(I,PSPACE)
C
C THIS ROUTINE SPECIFIES THE I COMPONENT OF THE CURRENT FIELD
C AT AN ARBITRARY POINT IN SPACE, PSPACE(I)
C
      COMMON /B1/ FEJUNC,IR,DELTAI,DELTA,IRS,TFJUNC,E,ES,FCAB,RCAB,JUMP,
     1PJUNCS,PCAB,PCABE,PCABO,RCABO,THETA,PJUNCO
      COMMON /B2/ NCAB,NNODE,ERJUNC,IRJUNC,DATA,DATN,H,PJUNC,
     1CDCAB,DCAB,FATE,NANC,ANJUNC,IREAD,IPRNT,INTAPE,OUTAPE,ITIME,IFLG,
     2OFLG,NIR,THETAS,THETAE,COMPD,THETAB,NJUNC,RHO,TEST,
     3NVSEG,ZVEL,VELZ,PIP,ECICAB,EXPCAB,ZJUNC,LJUNC,PATH,ICAB,IVOPT,
     4WCAB,IDEV,ICHECK,NDEV,NDATC
      DIMENSION FEJUNC(3,44),IR(3,44),IRS(3,44),TFJUNC(3,44),PJUNCO(3,44
     1)
      DIMENSION FCAB(3,51,22),RCAB(3,51,22),PJUNCS(3,44),PCAB(3,51,22)
      DIMENSION PCABE(3,51,22),PCABO(3,51,22),RCABO(3,51,22)
      DIMENSION NNODE(22),ERJUNC(44),IRJUNC(44),DATA(10),DATN(10),H(22)
      DIMENSION PJUNC(3,44),CDCAB(22),DCAB(22),ANJUNC(44),TEST(14)
      DIMENSION ZVEL(25),VELZ(25),ECICAB(22),EXPCAB(22),ZJUNC(22)
      DIMENSION LJUNC(22),PATH(22),ICAB(22),WCAB(22),IDEV(1000)
      DIMENSION ICHECK(44)
      DIMENSION PSPACE(3)
      INTEGER  OUTAPE,ZJUNC,ERJUNC,ANJUNC,OFLG
      INTEGER PATH
      REAL IR,IRS
      GO TO (10,10,30),I
   10 Z=PSPACE(3)
      DO 11 KK=1,NVSEG
      K=KK
      IF(Z.GT.ZVEL(K))  GO TO 11
      GO TO 12
   11 CONTINUE
      VFPS=(1.6878)*VELZ(K)
      GO TO 14
   12 IF(K.NE.1)  GO TO 13
      VFPS=(1.6878)*VELZ(K)
      GO TO 14
   13 SIGMA=Z-ZVEL(K-1)
      SLOPE=(VELZ(K)-VELZ(K-1))/(ZVEL(K)-ZVEL(K-1))
      VFPS=(1.6878)*(VELZ(K-1)+SLOPE*SIGMA)
   14 GO TO (21,22),I
   21 VELOC=VFPS*COS(THETA*PIP)
      RETURN
   22 VELOC=VFPS*SIN(THETA*PIP)
      RETURN
   30 VELOC=0.
      RETURN
      END
```

END

FILMED

4-86

DTIC